Simulating Challenges to Communication Networks for Evaluation of Resilience

Rabat Anam Mahmood

Submitted to the Graduate Degree Program in
Electrical Engineering & Computer Science and
the Graduate Faculty of The University of Kansas
School of Engineering in partial fulfillment of
the requirements for the degree of Master of Science

Thesis Committee:

Dr. James P.G. Sterbenz: Chairperson

Dr. Victor Frost

Dr. Güneş Ercal-Özkaya

01/20/2009

Date Defended

The Thesis Committee for Rabat Anam Mahmood certifies
that this is the approved version of the following thesis:

Simulating Challenges to Communication Networks for Evaluation of Resilience

Committee:

_____

Chairperson

_____

_____

_____

Date Approved

I would like to dedicate this research to my son, Aayan, whose bright sparkling eyes are my best motivation.

Abstract

This thesis is focused on the issues and challenges confronted by real network topologies. The goal of this thesis is to lead towards tools that permit network design to be as resilient as possible. In this thesis, a new capability has been introduced for the ns-3 simulator in which challenges such as failures, attacks, and natural disasters can be applied to different networks. Several simulations were performed to study the network resilience in the face of different challenges.

Acknowledgements

After graciously thanking Almighty God, I would like to express my gratitude to my advisory committee: Dr. James P.G. Sterbenz, Dr. Victor Frost, and Dr. Güneş Erçal-Özkaya. Thanks for hooking me up with this research. Special thanks to Dr. James P.G. Sterbenz for his time, patience, support, and understanding. Dr. Sterbenz, it has been an honor to work with you.

My gratitude also goes to the ResiliNets group. Special thanks to Abdul Jabbar, you were there to help no matter what time or day of the week. Justin P. Rohrer, your assistance was priceless. Egemen K. Çetinkaya, thanks for acting as a mentor to me. Asifuddin Mohammad and Qian Shi, I really enjoyed working with you.

The most special thanks goes to my best friend, my husband, who gave me his unconditional support and love all through this work. Hassan without you I would have lost my path. And above all, my parents, thanks for your faith in me that kept me going all this time.

Contents

List of Figures

Chapter 1

Introduction

This chapter focuses on the motivation behind this thesis and what eventually led to the idea of resilient networks. The motivation for this thesis leads to a problem statement that helps define the goals and objectives to be achieved by developing a challenge simulator to evaluate network resilience.

This chapter is organized as follows: First the existing situation of communication systems and the threats faced by them are described to outline the motivation behind this thesis. Second, the problem statement is clearly defined. Third, the goals and objectives for the simulator are stated. Finally, the contributions of this thesis are introduced.

1.1 Background and Motivation

Today we have an increasing reliance on computer networks and specifically the Global Internetwork. With emerging technologies, new applications and usage scenarios drive increased reliance on network services and infrastructure. This increasing reliance means that disruptions have a greater impact on society and

hence increasing attractiveness for those that want to harm society.

Furthermore, these networks are affected by natural disasters and human misconfigurations that are unintentionally induced in the networks. Natural disasters are inevitable whereas it is unrealistic to assure that human mistakes will not occur. In the event of an adversity leading to breakdown of the network, parts of the world can become isolated, reducing the utility of the network and its services. These severe consequences of disruption motivate the need for resilience to attack. Survivability of the network against adverse conditions is an essential aspect of dependable communications. Incorporating network resilience to threats and understanding the influence of challenges on network operation and performance is one of the key aspects of this thesis.

This thesis aims to provide tools that enhance resiliency and survivability of the network. "Resilience is the ability of the network to provide and maintain an adequate level of service in the face of challenges to normal operation", [34] whereas "survivability is the capability of a system to fulfill its operation, in a timely manner, in the presence of threats such as attacks or large-scale natural disasters. Survivability is a subset of resilience". [34]

1.2 Problem Statement

All communication networks are exposed to challenges such as natural disasters, attacks, or human mistakes. Therefore, the need arises to build communication networks that are resilient to such challenges. To analyse the behavior of existing and proposed networks under different scenarios, we need a framework that will simulate these challenges and the corresponding effects of the network under the induced challenge.

1.3 Goals and Objectives

The objective of this thesis is to design a framework that is capable of generating a challenge model, such as a natural disaster, that can be applied to a network model. The models are based on the user specifications, with challenge generation isolated from the type of network. Hence, any challenge can be imposed onto any network to observe its performance.

1.4 Contributions

This thesis provides a new tool that permits the separartion of challenge specification from the network model, so that resilience can be simulated without modification of the network model for each challenge. Furthermore this thesis demonstrates the utility of this technique by evaluating the resilience of several network topologies under several challenges.

1.5 Organization of the Thesis

The rest of the thesis is organized as follows: Chapter 2 provides an overview of the past research done that is the basis for the work done for this thesis. This begins with an overview of network simulators that can be used to simulate various real world scenarios. Resilience to challenges against the network is discussed with respect to various past research. Several existing topologies and topology generation tools are also discussed. Chapter 3 describes the diverse challenges that are confronted by the networks. Furthermore, the various aspects of resilience are introduced including different properties, attributes, and importance to understand the objective of this thesis. Chapter 4 discusses simulation implementation tech-

niques and describes the assumptions and abstractions along with the algorithms that were employed for the challenge simulation. Chapter 5 discusses the different simulation parameters that were essential to the testing of the simulator. Then simulation runs are described and analysed that produce several results to justify and hence prove the concept. Chapter 6 finalises the thesis with some conclusions and suggestions for future work .

Chapter 2

Background and Related Work

This chapter focuses on the different network simulation tools available for research along with a few examples of related work on resilience and survivability of the networks. After describing the functionalities of different simulators, the rationale of using ns-3 for this thesis is presented.

This chapter is organised as follows: First, several network simulators are discussed with special emphasis on the capabilities of each simulator relevant to the requirements of this thesis. Second, various network topology generation tools are discussed along with the way they differentiate from one another. Finally, some other related work is discussed.

2.1 Introduction to Network Simulators

To effectively engineer the Internet, proper understanding of its underlying structure is very important. Complete knowledge of the underlying structure is not possible due to its complexity and proprietary service provider topologies, therefore accurately mapping of its structure and evolution is not practical.

Hence for experimental and research purposes, some network simulators synthesise topologies that reflect many aspects of the actual Internet topology.

2.1.1 Network Simulators

Network simulators are frequently used for research into new protocols and architectures. Some of the important network simulators include the Optimized Network Engineering Tools (OPNET) Modeler, Global Mobile Information System Simulator (GloMoSim), Network Simulator 2 (ns-2), and Network Simulator 3 (ns-3).

2.1.2 OPNET Modeler Function and Capabilities

OPNET [21] Modeler is a commercial modelling and simulation tool that helps in designing and analysing communication networks. OPNET is a discrete event simulator with a user friendly graphical user interface (GUI). To aid research and development work in the field of communications, OPNET makes use of a wide variety of ready-to-use models, protocols, and technologies. Furthermore, OPNET also facilitates the design of custom models. This permits the user to either customise the existing models according to the requirements of the network or design completely new models.

Although OPNET is a very powerful simulation tool, a major drawback is its complexity. Customising and developing new models is not an easy task. Thorough understanding of the source code used in designing the models is required.

2.1.3 GloMoSim Function and Capabilities

GloMoSim [17] is a scalable simulation environment intended for wireless networks. The design of the simulator includes a parallel discrete-event simulation capability, and is built using a layered approach with standard APIs between the model layers. This helps to integrate models designed by different people. A visualisation capability is also present to see where and when the packets are being transferred, dropped, or rerouted. The platform used for designing models is Parsec [5], of which some knowledge is esstential. Parsec is a C-based simulation language developed for sequential and parallel execution of discrete event simulation models.

In spite of all its advantages, a major drawback of GloMoSim is the lack of a simulation environment for the wired networks. The capabilty for designing wired as well as hybrid wired-wireless networks is not available and hence the use of GloMoSim for this thesis was not possible.

2.1.4 Ns-2 Function and Capabilities

Ns-2 [19] is a network simulator that uses OTcl [16] as a command and configuration interface. OTcl is an object oriented version of Tcl. Ns-2 is an open source simulation tool and hence widely used by the academic and research community. There are quite a few important features and models present in ns-2, which include RED queue management, dynamic routing, multipath routing, two-way TCP, scheduling algorithms, and support for mobile hosts [19]. Furthermore, the output of simulations is readily available for observation via nam, the network animator [13]. This helps to actually visualise the flow of each packet within the network and hence understand the whole process.

Although these models are essential to any simulator, there are quite a few deficiencies in ns-2. Originally, ns-2 began with only wired network simulation functionality. Later, with the increasing need for simulating the wireless networks, wireless capabilities were added [1]. However, this is a different set of models that do not interoperate with wired models, so hybrid simulations are not possible. For every new release of ns-2, simulation models may have to be designed again as each simulation is tied to a particular release that makes ns-2 very fragile to changes. Furthermore, ns-2 is evolving rapidly to include new functionalities, models, and protocols, but unfortunately the proper documentation frequently does not keep pace.

2.1.5 Ns-3 Function and Capabilities

Ns-3 [20] is a completely new open-source, discrete-event simulator recently designed for the academic and research communities. As compared to ns-2, the distinguishing feature is a different configuration interface and set of scripts. Although the ns-3 simulator is written in C++ as is ns-2, the command and configuration interfaces are different. Ns-3 models are written only in C++ with some optional Python scripting. Hance, the scripts are not backward compatible and ns-3 is not an extension of ns-2. This permits a new modular design of network simulation models that can incorporate many more functionalities that were not present in ns-2.

Ns-3 contains inherent support for hybrid networks in which the type of each link, wired or wireless, can be specified. To observe and study the results from ns-3 simulations, pcap packet trace files are generated after the scripts are run; interpreting these files helps us to analyze the network's behavior. Ns-3 suffers

from a lack of models due to its recent development. If, on one hand, ns-3 lacks predefined models, on the other hand there are new functionalities important for this thesis.

There are certain generalisations used in ns-3. The basic computing device in ns-3 is represented by the **Node** class that provides methods for managing the devices in simulations. The user program is represented by the **Application** class that provides methods for managing the user-level applications in simulations. Nodes are connected to the network via the communication channel. The channel is represented by the **Channel** class that manages communication objects and connects them to nodes. All of the network devices and their respective software drivers, collectively called *netdevices*, are used to connect the computers to the network and are represented by the **NetDevice** class. This covers both the software driver and the simulated hardware part. The NetDevice class provides methods to manage connections to Node and Channel objects. Finally, to attach Nodes to NetDevices, Channels to NetDevices, and to assign the IP addresses, *topology helpers* are present. The topology helper creates NetDevices, adds MAC addresses, installs the NetDevice on the Node, configures the protocol stack on the node, and connects the NetDevice to a Channel, hence making the configuration setup easier. The modular structure of the ns-3 simulator makes it possible to easily deploy and analyse the network behaviors.

2.2 Network Topology Generation

To effectively engineer and evolve the Internet, a detailed understanding of its underlying structure is necessary. Mapping its actual topology is unfortunately a difficult task. However, topology generation tools have attempted to model the

modern Internet as a way to understand its behavior.

There are a wide variety of topology generation tools available that vary considerably in the way they synthesise the network.

2.2.1 Waxman

Waxman [35, 36] was among the first to develop an algorithm for the random generation of network topologies. The Waxman topology generator is a geographical model depicting the growth of the network. In this model, the nodes are uniformly distributed and are connected with one another based on a probability that is related to the distance between the respective nodes.

2.2.2 GT-ITM

The GT-ITM [18] topology generator attempts to reproduce the hierarchical structure of the Internet topology. It is a collection of software tools for creation, manipulation, and analysis of graph models of the topologies. It also includes enhanced visualisation capabilities, a routing and forwarding module for use with large graphs, and support for modeling of interdomain routing policies.

2.2.3 BRITE

BRITE [29] is a universal topology generation tool designed to be flexible to operate and user friendly. This tool is extensible; the functionalities can be further enhanced.

A distinguishing feature is that BRITE supports both flat and hierarchical topologies. Furthermore, it allows interoperability between topologies from other topology generators. BRITE models can be assigned link attributes such as band-

width and delay. It also allows the user to provide custom configurations for the topology.

2.2.4 KU-LoCGen

The University of Kansas location and cost constrained network topology generator (KU-LocGen) [28] is a tool to enable geographic node positioning and the respective cost constraints on the topologies generated by well-known graph generation algorithms. These two features greatly enhance the utility of the existing topology generators.

Most of the topology generators focus upon the generation of interconnecting links between the nodes irrespective of the node position. Nodes are generally placed in a random fashion that does not depict real-world scenarios. This is important as network designers are generally constrained by the position of the nodes.

Capital constraints are a significant limiting factor in network deployment. KU-LoCGen uses a realistic cost function to determine a range of affordable model parameters that provide feasible topologies that optimise the performance under cost constraints.

These two features prove to be quite important for this thesis. Using the node positioning facility, the real networks under consideration can be accurately and precisely positioned, whereas the cost constraints dictate the link structure of a network. KU-LoCGen creates different topologies which include purely random, locality based, and Waxman models. The topologies used for this thesis were those generated using the Waxman model.

KU-LoCGen incorporates network design practices in topology generation,

thereby enabling a tool that can be used to generate and analyse viable alternate topologies during the network design and engineering phase.

2.3 Other Related Work

Geol, Belardo and Iwan proposed a self-healing and self-managing network [27], the purpose of which was to provide uninterrupted communication between government agencies when part or all of the network goes down because of any mishap or crisis. They took the motivation from the drawbacks of the communications network which became evident after the 9/11 attacks on the World Trade Center. The architecture proposes independent services with standard interfaces and variable addresses. Under any sort of a challenge, the services that are affected are segregated from the network and the redundant, independent services replace them at alternate nodes. The redundant services discover each other by matching the standard interfaces and allow even the complex operations to be performed through them. Furthermore, these new services add to the redundancy of the network at times of network overload. This means the new services will not allow the performance of the network to degrade due to traffic overload.

Xiangqian, Makki, Kang, and Pissinou proposed Adapting Traffic Evolution Topology gEnerators (ATETEs) which provide three types of network backbone evolution methods [24]. This corresponds to topology generation but under stressed network conditions. The need for network backbone evolution arises when the traffic through the network increases and exceeds the capacity of the network infrastructure. The three main types of evolution methods proposed include link upgrading, node upgrading, and their combination. In the link upgrading scenario, link capacity is increased whereas in the node upgrading scenario, more nodes and

links of the same link capacity are added. ATETEs consider traffic distribution and select the congested link as the main shunting object. Compared with most of other topology generators such as BRITE that do not consider traffic, ATETEs are more efficient to satisfy traffic increase.

Chapter 3

Challenges and Resilience

This chapter focuses on resilience and survivability of networks, along with different challenges faced by the communication networks.

This chapter is organized as follows: First, several resiliency traits and strategies are discussed. Second, different kinds of challenges that impair the normal working behavior of the networks are categorized and their respective effects are classified. Finally, an introduction is provided to the way in which challenges are modeled for this thesis.

3.1 Resilience

"*Resilience* is the ability of the network to provide and maintain an acceptable level of service in the face of various faults and challenges to normal operation" [8]. A resilient network detects, remediates, and recovers from outages and periods of degraded performance within as little time as possible. This so happens that each node in the network continuously monitors the functioning and quality of the associated paths. The information hence gathered is analysed by the node

itself. Based on the condition of the paths, the nodes decide the best suited path to route the packets. This whole process aims to enhance the overall resilience of the network.

The diagram in Figure 3.1 shows the characteristics that make a system resilient.

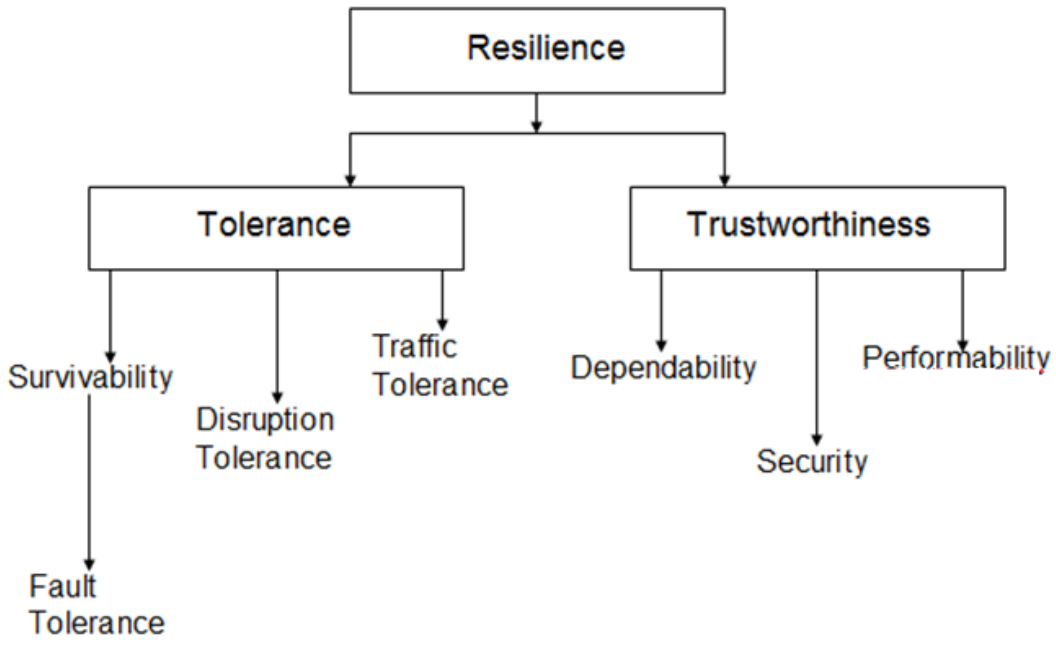


Figure 3.1. Resilience Overview

3.1.1 Tolerance

The first major aspect of resilience is tolerance. "*Tolerance* refers to the ability of the communications network system to endure service failures". It is known from experience that failures are inevitable, in part because it is not possible to design a perfect system. There are several important aspects of tolerance,

described in the following subsections.

3.1.1.1 Fault tolerance

"*Fault tolerance* is the ability of a system to tolerate faults such that service failures do not result". Faults can be treated as random and independent events; either single or very few in number [4, 22]. Generally, systems are made fault tolerant by eliminating every known failure and by providing redundancy to permit continued operation even when failures do occur. In this thesis, fault tolerance with link redundancy is employed as the basis for analysis of resilience.

3.1.1.2 Survivability

"*Survivability* is the capability of a system to fulfil its mission, in a timely manner, in the presence of threats such as targetted attacks or large-scale natural disasters resulting in many failures, in addition to the few random failures covered by fault tolerance" [11, 25]. In other words, it refers to the ability of the communications network system to operate as intended, even if the challenges are considerably more severe than independent random failures. This thesis works on the principle of survivability, particularly when the network is under attack or is impaired because of some operational or component failure.

3.1.1.3 Disruption tolerance

"*Disruption tolerance* is the ability of a system to tolerate disruptions in connectivity among its components" [3, 26]. It covers disruptions from environmental challenges including weak and episodic channel connectivity, mobility, and delay.

3.1.1.4 Traffic tolerance

"*Traffic tolerance* is the ability of a system to tolerate unpredictable offered load without a significant drop in carried load (including congestion collapse), as well as to isolate the effects from cross traffic, other flows, and other nodes" [15]. The traffic can either be unexpected but legitimate such as from a flash crowd, or malicious such as a DDoS attack. Traffic patterns usually vary and most of the times they are unpredictable.

3.1.2 Trustworthiness

The second major aspect of resilience refers to properties of dependability, security, and performability that can be measured to analyse how the nature responds to challenges. "*Trustworthiness* relates to something which is worthy of being trusted to satisfy its specified requirements, in the presence of a wide range of adversities" [30]. There are a number of aspects of trustworthiness, described in the following subsections.

3.1.2.1 Dependability

"*Dependability* is the property of a system such that reliance can justifiably be placed on the service it delivers" [2]. It generally includes the measures of availability (ability to use a system or service) and reliability (continuous operation of a system or service), as well as integrity, maintainability, and safety [23].

3.1.2.2 Security

"*Security* is the property of a system and measures taken such that a network is capable of protecting itself from unauthorised access or change, subject

to policy" [10]. Security can be broadly thought of as encompassing everything that promotes the prevention, detection, and remediation of deleterious system behavior which includes actions of people, information technology, and physical environments [30]. Typical properties include AAA (auditability, authorisability, authenticity), confidentiality, and nonrepudiality. Security shares with dependability the properties of availability and integrity.

3.1.2.3 Performability

"*Performability* is the property of a system such that it delivers performance required by the service specification, as described by QoS (quality of service) measures" [6]. The goal is for systems to provide continued performance even when challenged. Although the performance level declines, the system is still functional and provides logically correct operation. This is made possible in part by the use of redundant components to permit the system run [7]. In this thesis, when a certain failure occurs in the network, the nodes recalculte to figure out a way to route the packets through some other paths to provide performability. Performability in terms of goodput and packet delivery ratio is the measure of resilience used in this thesis.

3.2 Challenges

A challenge can be defined as an adverse event or condition that impacts the normal operation. Different challenges cause different type of errors, some of which are discussed below.

3.2.1 Flash Crowds and Denial of Service Attacks

Flash events and denial of service are the two common challenges. Both of these overwhelm the network resources to a point that either the services are degraded or completely fail.

A flash crowd is a huge surge in legitimate traffic to a single point such as a web server causing a significant increase in the network load resulting in packet loss and congestion.

Denial of service (DoS) attacks are malicious challenges that saturate network resources so that the services cannot be provided to the intended users. When a network is under a DoS attack, an attempt can be made to distinguish and isolate the malicious traffic.

3.2.2 Challenges in Wireless Networks

Mobile wireless networks face disruptions from a number of factors. A major source of disruption is mobility of network nodes. Routes change due to movements and often result in disconnects. Wireless channels are generally noisy, weak, and asymmetric. Furthermore, there may be unpredictably long delays either due to long satellite links or episodic connectivity.

3.2.3 Attacks

Attacks can directly target either the software, the hardware, or the protocol infrastructure of the communication network in an attempt to disrupt the communications and make the services unavailable. Attacks can be launched by recreational crackers or terrorists.

3.2.4 Misconfigurations

Human misconfigurations and operational errors are not uncommon. This may include a firewall blocking legitimate traffic, a load balancer unable to use all resources, or routing protocols announcing the wrong prefix route combination resulting in traffic following the wrong path.

3.2.5 Natural Faults

Natural faults result from design flaws or network component aging. These are random failures and quite inevitable.

3.3 Simulated Challenges

The different challenges simulated and hence studied for this thesis are discussed in this section.

3.3.1 Random Faults

Random failures result from component failures and non-malicious human error such as fiber cuts.

3.3.2 Natural Disasters

Natural disasters such as hurricanes are a major threat to communication network that has geographical scope. To simulate this scenario we can define an area A over which the challenge is applied to the network. This area can either be an n-sided polygon $(x_0\ ,\ y_0)$, $(x_1\ ,\ y_1)$, $(x_{n-1}\ ,\ y_{n-1})$ or a circle with center at x_0, y_0 and radius r.

3.3.3 Attacks by Intelligent Adversary

An intelligent adversary attack is characterised by the failure of components that may impact the network the most. This is the case when an attacker has knowledge about the network's physical topology.

Chapter 4

Simulation Implementation

This chapter describes the simulation environment, the simulation code, algorithms, and protocols used to overcome the design problems.

This chapter is organized as follows: First an overview of the simulation environment is provided. Second, the code implementing the challenge simulation is explained highlighting the critical aspects. Third, an overview of the algorithms used in the simulation code is provided. Finally, this chapter discusses the KU-LoCGen topology generator, which is used as an input to the analysis presented in Chapter 5.

4.1 Introduction to the Simulation Environment

Ns-3 is the simulator used for this thesis as described in section 2.1.5.

4.1.1 Abstractions and Conceptual Overview

As discussed earlier, there are certain abstractions used in ns-3. The basic computing device abstraction represented by the **Node** class provides methods

for managing the devices in simulations. The abstraction for the user program is represented by the **Application** class that provides methods for managing the user level applications in simulations. Nodes are connected to the network via the communication channel. The channel is abstracted by the **Channel** class that manages communication objects and connects nodes to them. There are specialised channels present in ns-3 that include CsmaChannel, PointToPointChannel, and WiFiChannel. All of the network devices and their respective software drivers collectively called *netdevices* are used to connect the computers to the network and are abstracted by the **NetDevice** class. This covers both the software driver and the simulated hardware. The NetDevice class provides methods to manage connections to Node and Channel objects. Finally, to attach the Nodes to the NetDevices, the Channels to the NetDevices, and to assign the IP addresses, topology helpers are present. Topology helpers create NetDevice, adds MAC address, installs the NetDevice on the Node, configures the protocol stack on the node, and connects the NetDevice to a Channel hence making the configuration setup easier.

Several different topology helpers are used in ns-3 topology generation. The NodeContainer topology helper provides a convenient way to create, manage, and access Node objects that are created in order to run a simulation. The NetDeviceContainer helps to manage and access device objects. The PointToPointHelper assists in the configurations and connections between a PointToPointNetDevice to the PointToPointChannel. The InternetStackHelper installs an Internet Stack (TCP, UDP, IP, etc.) on each of the nodes in the NodeContainer. The IPv4AddressHelper manages the allocation of IP addresses to the device on the node, from 10.1.1.0 with netmask 255.255.255.0. By default the

addresses allocated start at 10.1.1.1 and increase monotonically.

4.2 Simulation Code Organisation

The challenge simulator organisation takes a few inputs from the user which include, in a broad sense, the network descriptor files and challenge descriptor files.

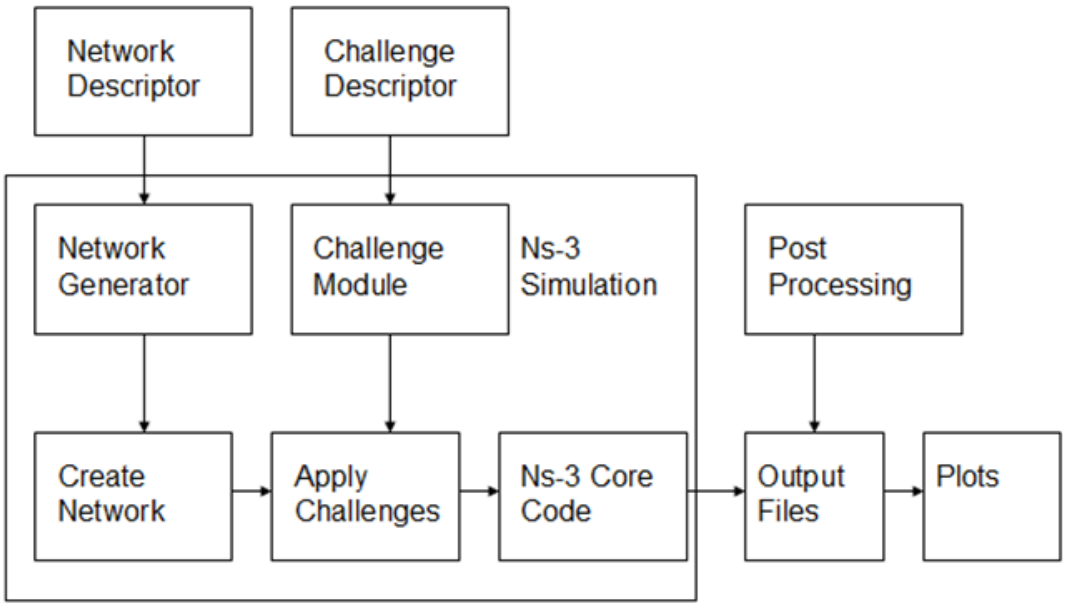


Figure 4.1. Simulation Flow

The network descriptor files contain data regarding a specific network topology whereas the challenge descriptor file contains the specifications of the challenge to be imposed onto a network. A significant contribution of this thesis is that the network description has been separated from the challenge description. This is to say that any sort of a challenge can be applied to any communications network architecture for the purpose of studying it's resilience under attack. This reduces the problem to c challenge descriptors and n network models rather than $c \times n$

combined models.

Figure 4.1 shows the overall organisation of the challenge simulation process.

4.2.1 Network Descriptors

The simulator generates the network using two different types of network specification files namely, the node coordinate matrix and adjacency matrix.

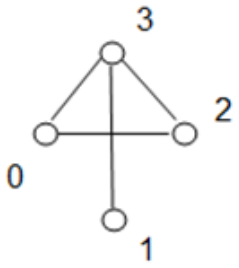
The network specification file is the node coordinate matrix file that specifies the geographical location of the nodes in the form of x-coordinates and y-coordinates as shown in Figure 4.2. The location of each node is an important property to include in case of a geographical failure resulting from a natural disaster. Nodes that fall within the region of impact are put down by the simulator.

The adjacency matrix file contains all the adjacent interconnections between different nodes of the network and this defines the links. Figure 4.3 shows example nodes that are interconnected and the corresponding matrix. Once the links are established they are shown by 1's in the matrix. Using this file the simulator creates IPv4 associations between the nodes, which permits the simulator to determine which links will be affected by a given challenge. This file also provides the simulator with the total number of nodes in the network.

4.2.2 Network Topologies

Several network topologies are generated using KU-LoCGen, introduced in section 2.2.4. The actual topology adjacency matrix is based on the real network deployed by a network operator. Figure 4.4 shows the actual Sprint router-level topology as inferred by Rocketfuel [33].

For the sake of comparison, additional topologies based on Sprint node loca-

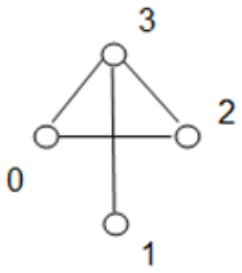


	X-coordinate	Y-coordinate
0	50	10
1	60	20
2	70	30
3	0	40

Figure 4.2. Example Node Coordinate Matrix

tions were generated by KU-LoCGen. The *synthetic resilient* topology has richer interconnections and hence lower probability of service disruption under failures. Figure 4.5 shows the Sprint synthetic resilient topology. Increased resilience with more redundant links result in a significantly increased cost.

The third scenario is the *synthetic fragile* topology, which is less topologically interconnectedand is therefore less resilient. Ideally a network engineer should be able to determine the right balance between connectivity (resilience) and cost. Figure 4.6 shows the fragile Sprint topology. Note that even a single node or link



	0	1	2	3
0	0	0	1	1
1	0	0	0	1
2	1	0	0	1
3	1	1	1	0

Figure 4.3. Adjacency Matrix

failure near the middle of the graph can partition the network.

Additionally, the European research network GÉANT2 [12] topology is studied for comparison. Figure 4.7 shows the GÉANT2 actual topology.

4.2.3 Challenge Descriptors

Challenge description is a critical aspect of this thesis. The challenge descriptor file provides the necessary information needed to put the network under attack and hence observe its resilience. This thesis does not directly provide a means of

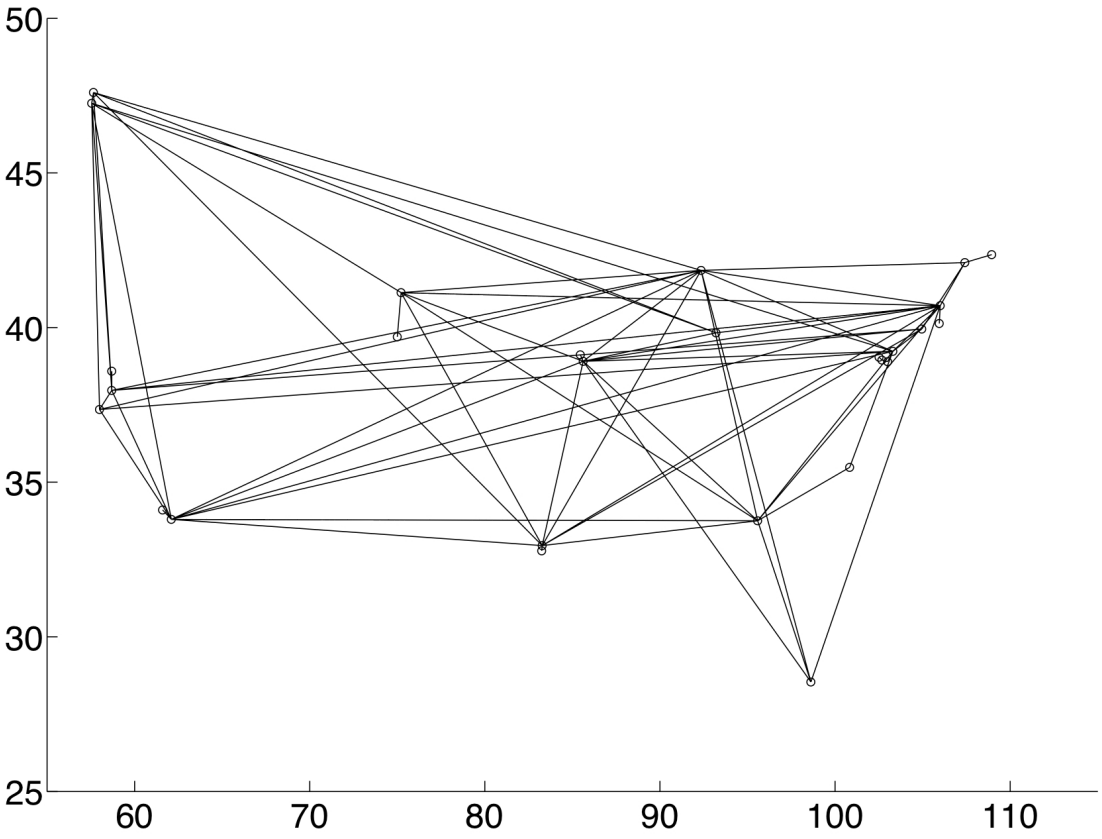


Figure 4.4. Actual Sprint Topology

improving the resilience of the network, but rather provides tools to help study the level of resilience for a certain network under different challenges.

This simulator design focuses on four types of challenge scenarios which include randomly occuring failures, geographical failures, link failures, and intelligent adversary attacks.

4.2.3.1 Random Failures

In case of random failures, the user provides the number of nodes which are to be shut down as input to the simulator. Using a random number generator, the simulator randomly selects nodes and shuts them down at the time specified in the

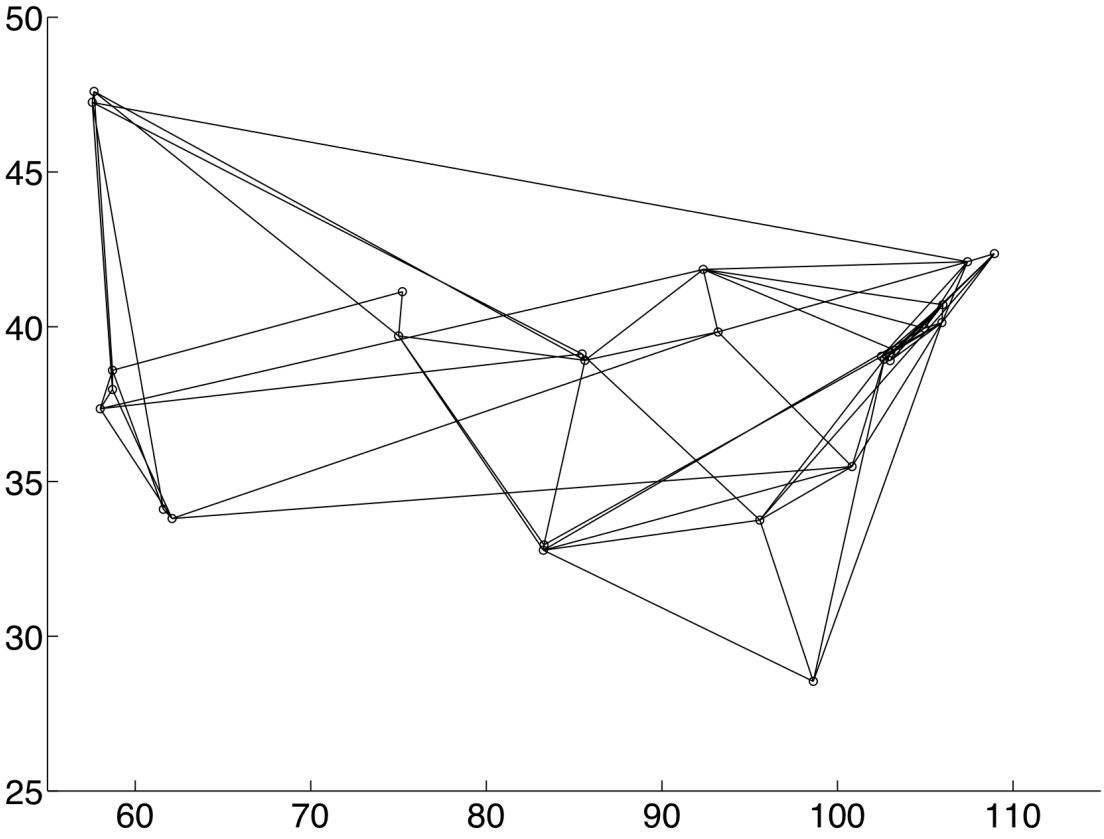


Figure 4.5. Sprint Synthetic Resilient Topology

challenge descriptor file. After being shut down for a specific duration the network regains the nodes, if and when the challenge is removed. Hence, the behavior of the network when some operational failure occurs can be studied and the system can be modified accordingly. In this thesis, five runs are averaged with a different set of a given number of random nodes down.

4.2.3.2 Geographic Failures

In an area-based challenge, the user provides the area specifications in the terms of geographic coordinates (longitude, latitude) to the simulator. The simulator uses the ray casting algorithm [32] to figure out how many and which nodes

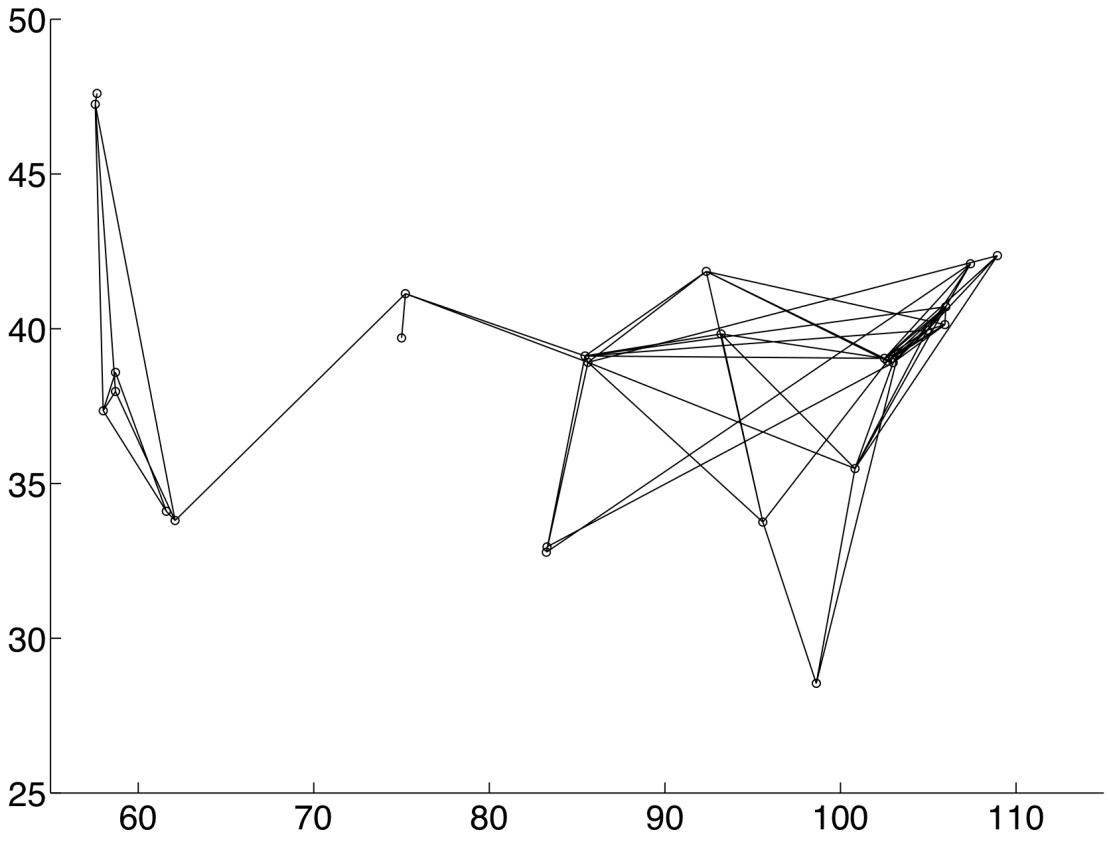


Figure 4.6. Sprint Synthetic Fragile Topology

lie under the direct impact of the challenge and will be cut off from the rest of the network during a geographical failure. A ray is initiated from any node in one direction and extended untill the end of the simulation region. The number of times this ray cuts the challenge polygon boundary is counted. If the number of crossings is an even number, the node lies outside the affected region, as in Figure 4.8. If, however, the number of crossings are odd, then the node lies inside and will be shutdown, as in Figure 4.9. The simulator then shuts nodes within the polygon down at the time specified by the user, which generally represents a natural disaster such as a hurricane.

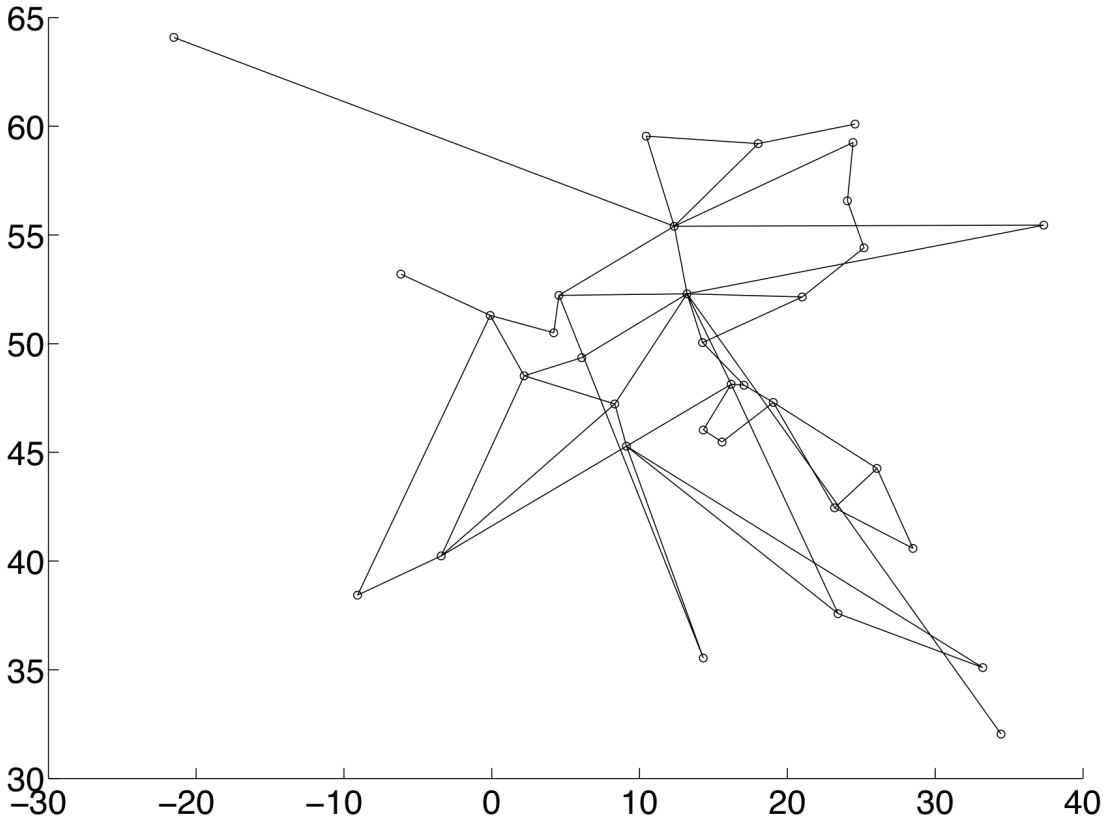


Figure 4.7. GÉANT2 Topology

4.2.3.3 Link Failures

Optical fiber cable links are the most reliable and widely deployed means of long-haul communications in the world today. Accidental fiber cuts can impact network's operation. The Link Cut model is be used to simulate a random cable cut in networks.

4.2.3.4 Attack by an Intelligent Adversary

A significant threat to normal operation is an attack by an intelligent adversary who has knowledge of the network topology. Key nodes or links may be targetted to inflict major damage. This simulator has the ability to put down certain

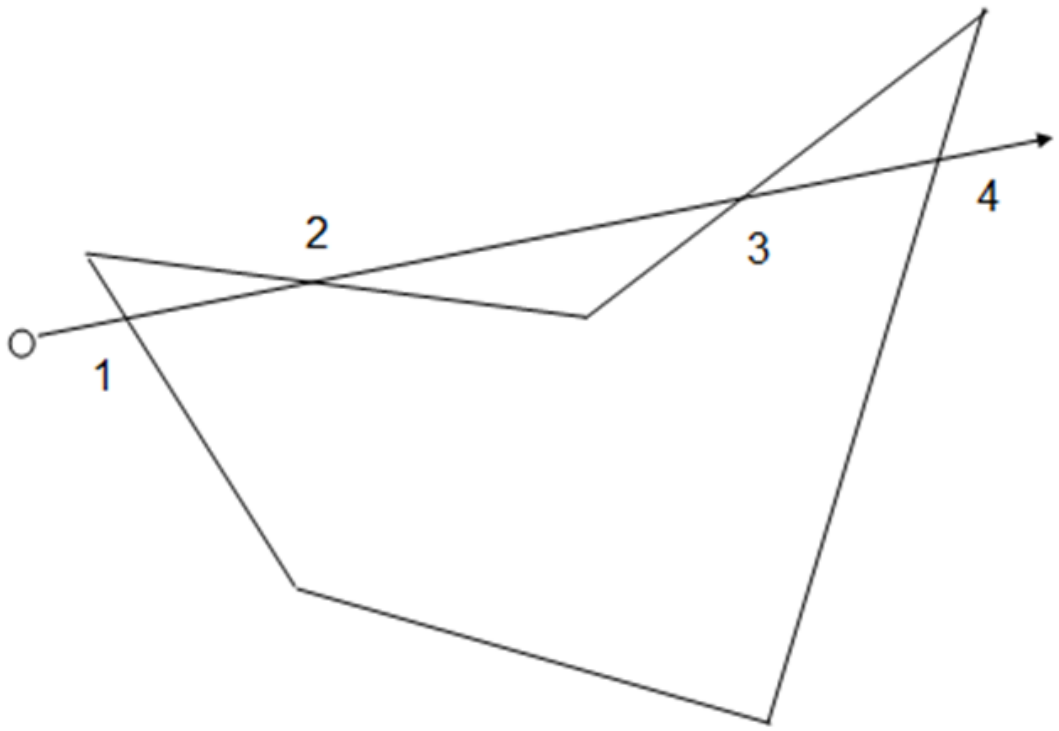


Figure 4.8. Ray Casting Algorithm Example (Even)

nodes or links to analyse network performance under targetted attacks. User can provide a list of nodes to be shut down or a list of links that are intended to be targetted. To better engineer the cable connections and node interconnections, the user observes the behavior of the network once those nodes and links go down and come up again.

4.3 Simulation Models

An ns-3 C++ object is created for each instance of the network. Each connected node pair is kept in a node container that helps in managing and accessing the respective node objects. Since this simulation code deals only with the wired

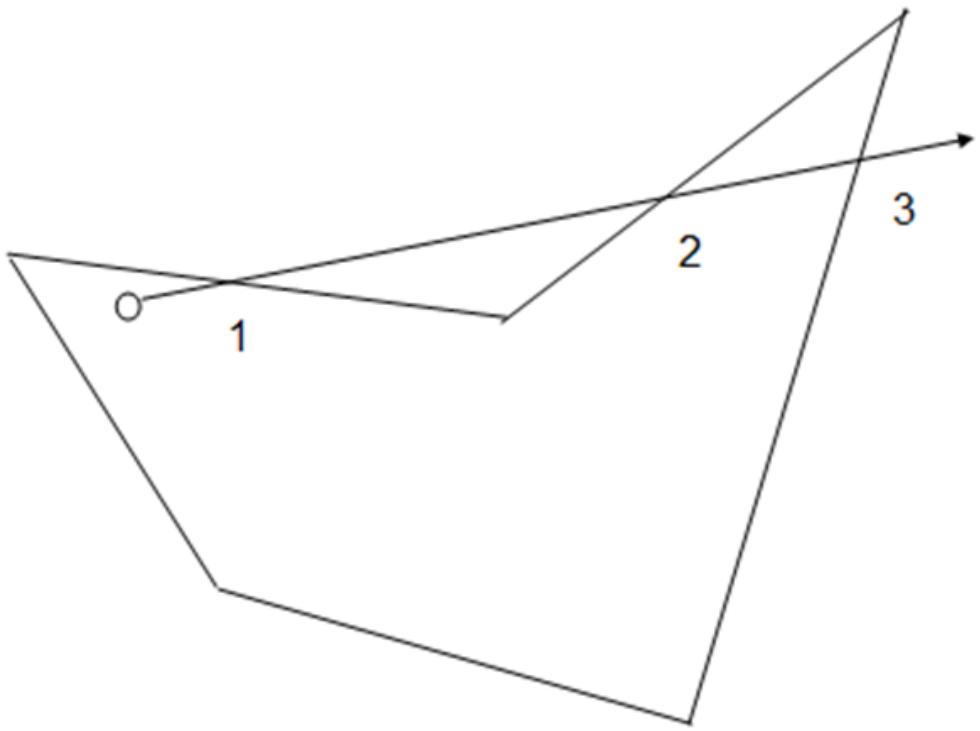


Figure 4.9. Ray Casting Algorithm Example (Odd)

networks, nodes are fixed at a certain geographic location. Nodes are interconnected via a full-duplex point-to-point link. In order to make the link full duplex, there are two wires in the channel each of which have an IDLE and a TRANSMIT state. Each link has a capacity of 5 Mb/s and imposes a 2 ms transmission delay for each packet. In ns-3, the value of delay, frame size, and the inter-frame gap can be assigned to model a real network.

In the simulation code, different IP addresses have been assigned to each interface representing a mesh of individual nodes. For experimental purposes, each network node has one application node attached to it. This makes a total of 27 application nodes for Sprint and 34 application nodes for GÉANT2. Each appli-

cation node generates traffic destined for the rest of the application nodes. Hence, each application node is a source as well as a destination for the traffic flowing from all the other nodes. The traffic flowing from the application nodes in the network is constant bit rate (CBR) at 40 kb/s with a packet size of 1000 bytes. The network protocol used for the simulator is UDP (user datagram protocol). UDP allows network applications to send messages to another host without requiring to setup a connection between the two hosts apriori. UDP is an unreliable service because no handshaking and resource allocation is completed before the traffic starts to flow in the network. This unreliability often causes packets to be received out of order, duplicated or to go missing without notice.

For this thesis, the routes are statically configured using the Dijkstra's shortest path algorithm. During route computation nodes share information that is used to approximate link state routing. However, in the event of a failure, the routes are manually recomputed to give the effect of adaptive routing.

Chapter 5

Results and Analysis

This chapter aims to justify the simulator design with the help of several simulations and their results. Various challenge scenarios were designed and applied onto the given networks to evaluate the simulator's concept and functioning.

This chapter is organized as follows: First, some of the relevant performance metrics are discussed. All the results are studied in terms of *goodput* and *packet delivery ratio*. Second, the parameters used in this simulation are described. Next, a variety of failure scenarios are tested and plotted. Finally, some comparisons are made across topologies that give some initial insight into the benefits of the challenge simulator.

5.1 Performance Metrics

To evaluate the proper functioning of the simulator design and to justify the goals and objectives which were intended from this thesis, certain performance metrics were studied. To study the resilience of networks, the most relevant metrics for this thesis were goodput and packet delivery ratio, which reflect the

performability of the network under challenge.

5.1.1 Goodput

The rate at which useful data is received at the destination is referred to as the goodput. Goodput calculation leaves out packet headers and information lost or corrupted in transit along with any duplicate transmissions or retransmissions. Goodput can be thought of as throughput seen by the receiver.

5.1.2 Packet Delivery ratio

Packet delivery ratio refers to the ratio of the number of packets successfully received at the destination to the total number of packets sent by the sender.

5.2 Simulation Parameters

One of the essential parts of justifying the simulator design includes its thorough evaluation. To test the simulator, various parameters are chosen and the performance measured accordingly.

The simulator has the capability to separate the core nodes of the network from the application nodes. The number of core nodes and their respective positions are obtained from the network description files whereas the application nodes and their respective locations can be variable. For evaluation purposes, one application node is attached to each core node and an arbitrary position is assigned to it.

Link bandwidth or data rate that refers to the total link capacity is chosen to be 10 Mb/s both for the core and the application nodes to avoid any bottlenecks. A *delay* of 2 ms is set for the links which depicts how quickly the packet delivery will take place.

To assign IP addresses to the nodes two IP pools are created, one for the core nodes and the other for the application nodes. As the simulator generates the topologies, it extracts one IP address from the core pool and one from the application pool and assigns them to the respective nodes. The network portion of the IP addresses are also arbitrarily chosen: 10.1.0/24 for the core and 192.1.0/24 for the application nodes.

The networks are sufficiently overprovisioned so that traffic is not dropped when there are no challenges. UDP (User Datagram Protocol) packets of size 1000 bytes are sent at a rate of 40 kb/s and the results observed.

To calculate the number of packets sent per second per node (PPS), we have the following formula:

$$\text{PPS} = \frac{nft}{8B}$$

where n corresponds to the total number of nodes present in the network, f corresponds to the number of flows per node, t corresponds to the rate at which the data is sent measured in bits per second, and B corresponds to number of bytes sent per packet.

The simulation runs all start at 0.0001 seconds and stop at 15.0001 seconds with two seconds given at the end to shut down all the processes before the simulator stops completely. The challenge is applied based on the challenge specification file, which for these simulations is from 5.0000 seconds to 10.0001 seconds.

At first the simulator is tested without any challenge applied to it. After receiving the desired results, the simulator is tested against the challenge scenarios. For each scenario several iteration are carried out to average the results. The random cases are averaged over five runs to get a nominal value.

5.3 Failure Scenarios

Fiber optic cable cuts and node failures can have a significant impact on network operation hence their study is essential. Both natural failures and operator accidents are inevitable and can be treated as random effects. Hence, we studied the network behavior under various random link and random node outages.

5.3.1 Link Failures

In this section, we examine the impact of link failures on performability. For the GÉANT2 actual topology with 34 nodes and the Sprint actual topology with 27 nodes, the results were as expected. Equal number of random links were shut down for each of the networks with results averaged over five sets of failures. Figures 5.1 and 5.2 show the results for the GÉANT2 actual topology with 1, 2, 3, 4, 5, 10, 20 links put down. The graphs clearly show more degradation of the throughput as the number of links down increases. Similarly, the packet delivery ratio decreases with increasing number of links down.

Figure 5.3 and 5.4 show the results for Sprint actual topology under random link outages. The plots appear to show less degradation as compared to the GÉANT2 actual topology even for 20 links shut down. This is because GÉANT2 has fewer redundant paths than the Sprint topology, however traffic has not been normalised to permit direct comparison between these two topologies. Similarly, the packet delivery ratio decreases with increasing number of links down, but not as much as for the GÉANT2 topology.

Similar experiments were performed on the Sprint synthetic resilient and fragile topologies to further justify the simulator's functionality. The Sprint synthetic resilient topology showed results similar to Sprint actual topology, while the Sprint

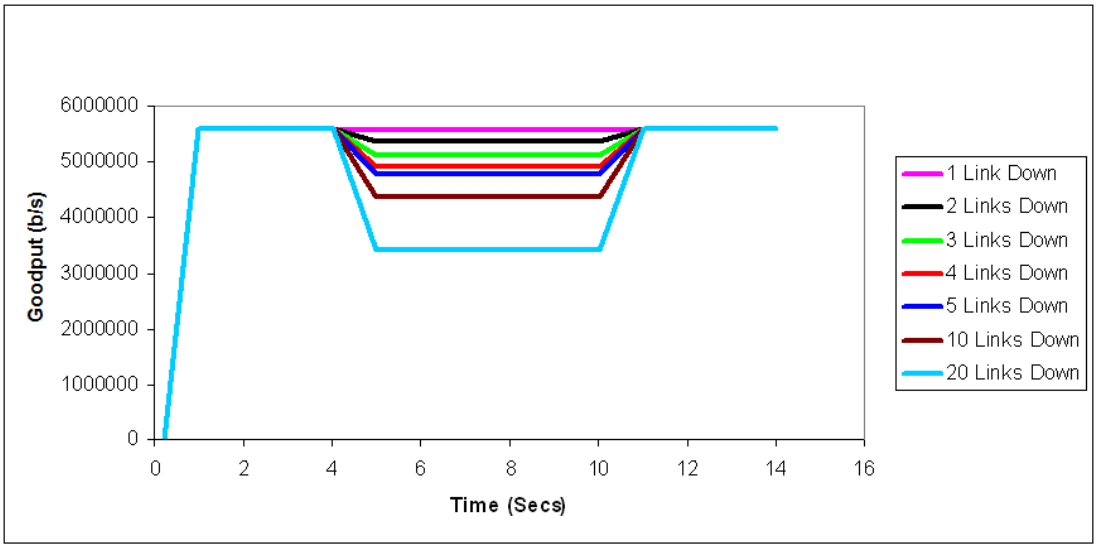


Figure 5.1. Random Link Failures of GÉANT2 Actual Topology

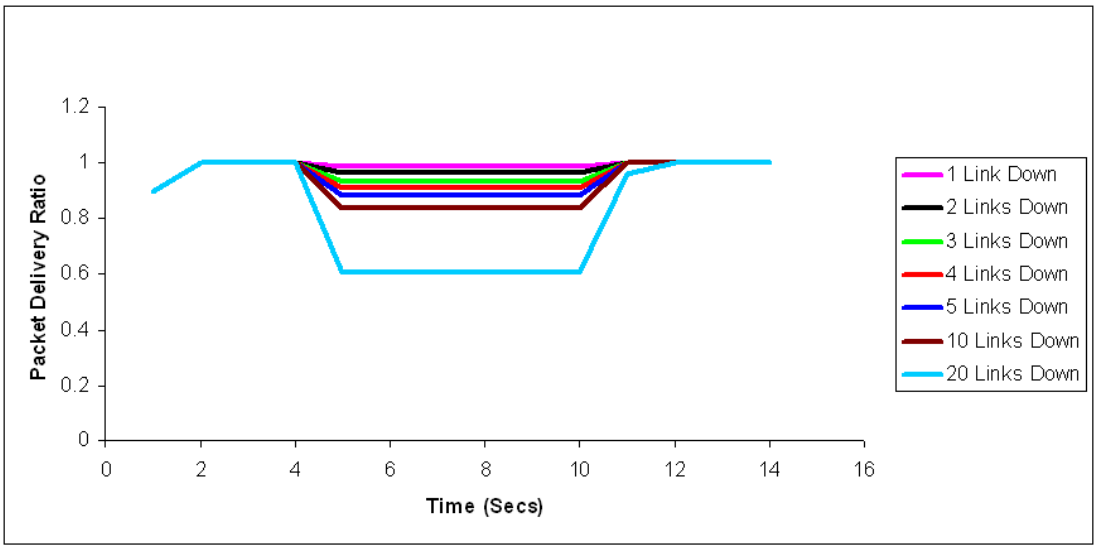


Figure 5.2. Random Link Failures of GÉANT2 Actual Topology

synthetic fragile topology did show significant degradation depicting network segregation.

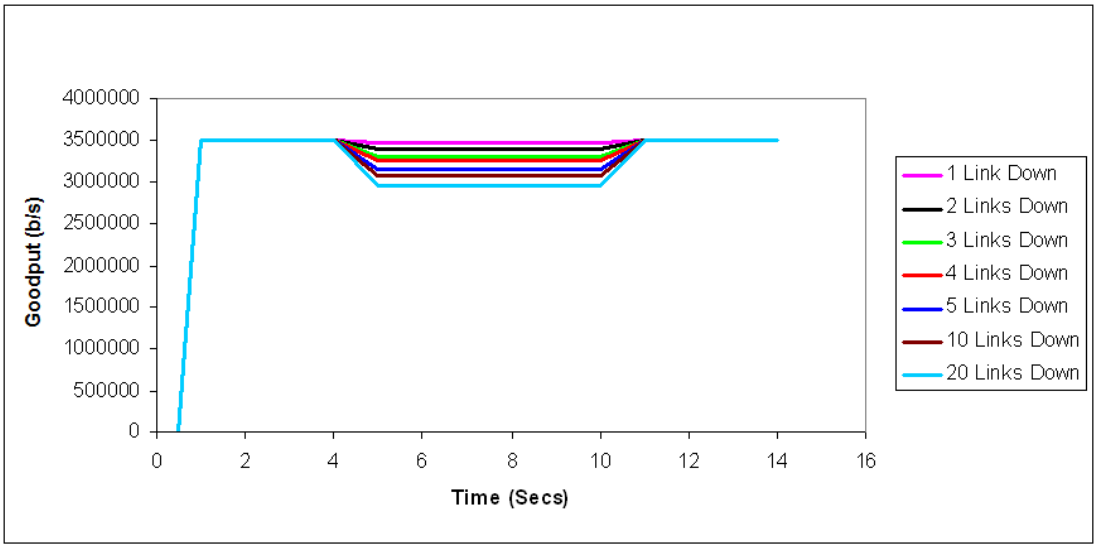


Figure 5.3. Random Link Failures of Sprint Actual Topology

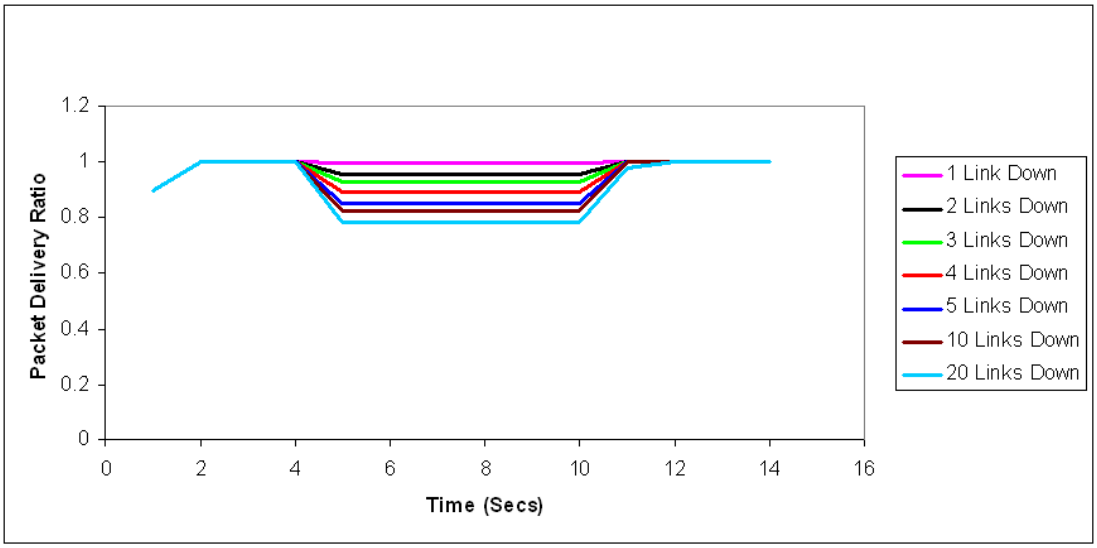


Figure 5.4. Random Link Failures of Sprint Actual Topology

5.3.2 Node Failures

In this section we examine the effects on performability of node failures. For the GÉANT2 and Sprint actual topologies, the results were also as expected. An equal number of random nodes were shut down for each network. Figures 5.9

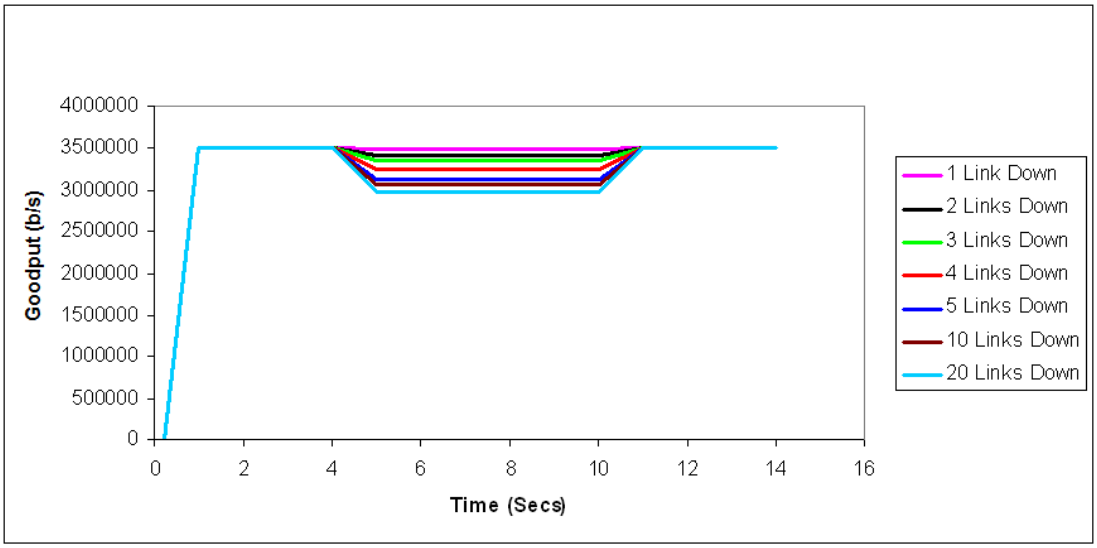


Figure 5.5. Random Link Failures of Sprint Resilient Topology

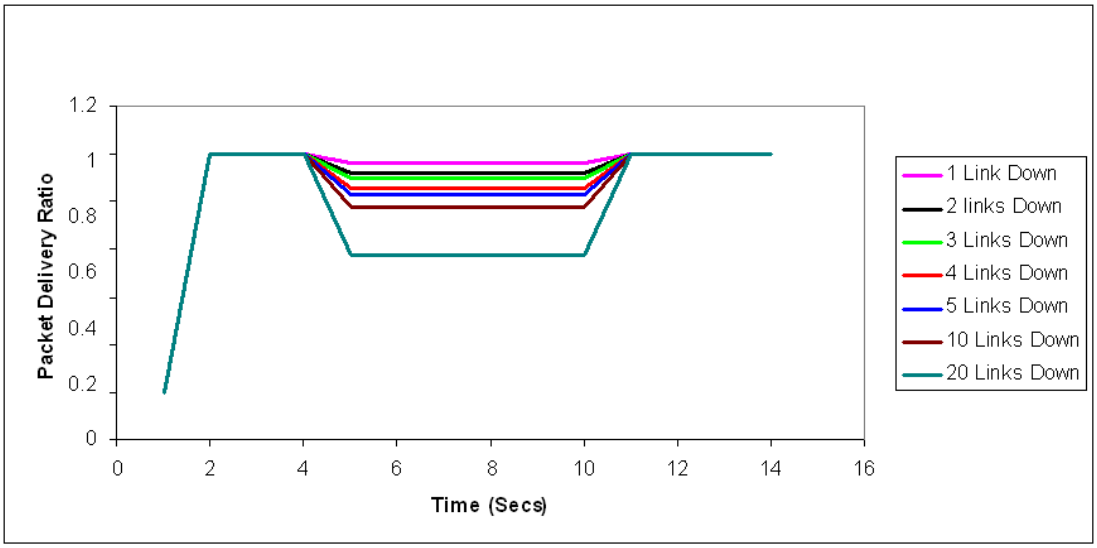


Figure 5.6. Random Link Failures of Sprint Reslilient Topology

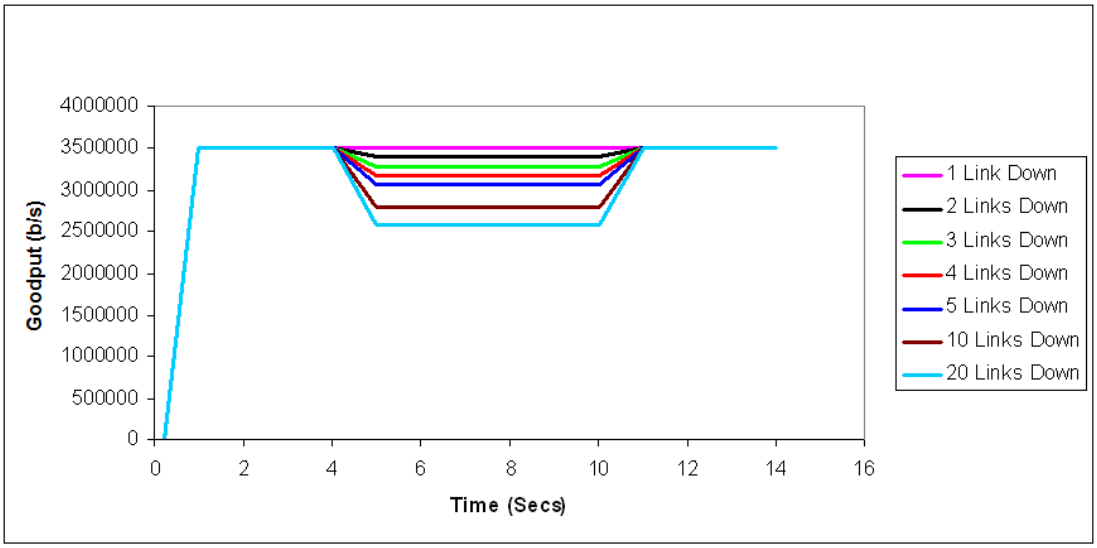


Figure 5.7. Random Link Failures of Sprint Fragile Topology

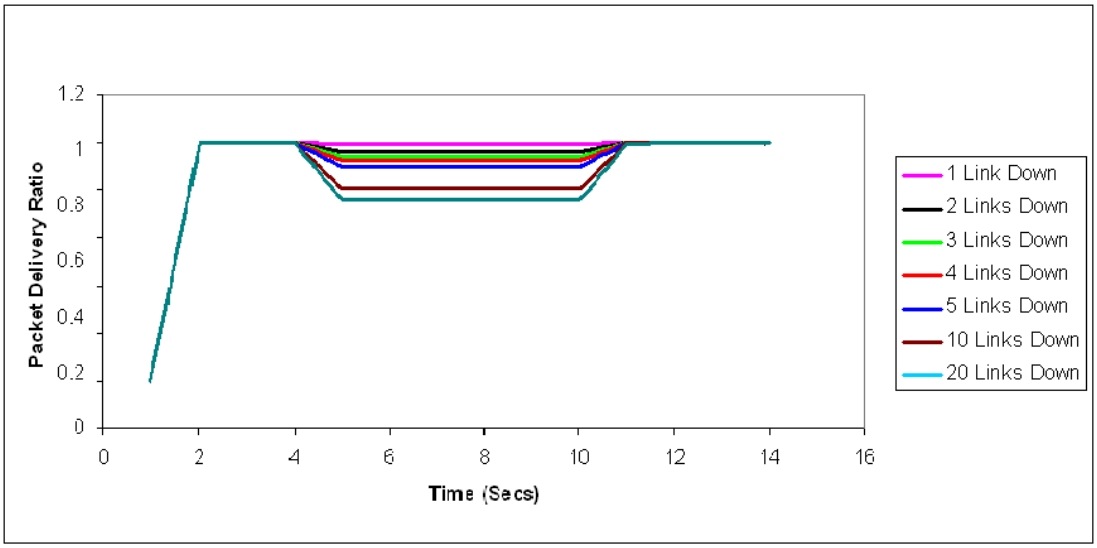


Figure 5.8. Random Link Failures of Sprint Fragile Topology

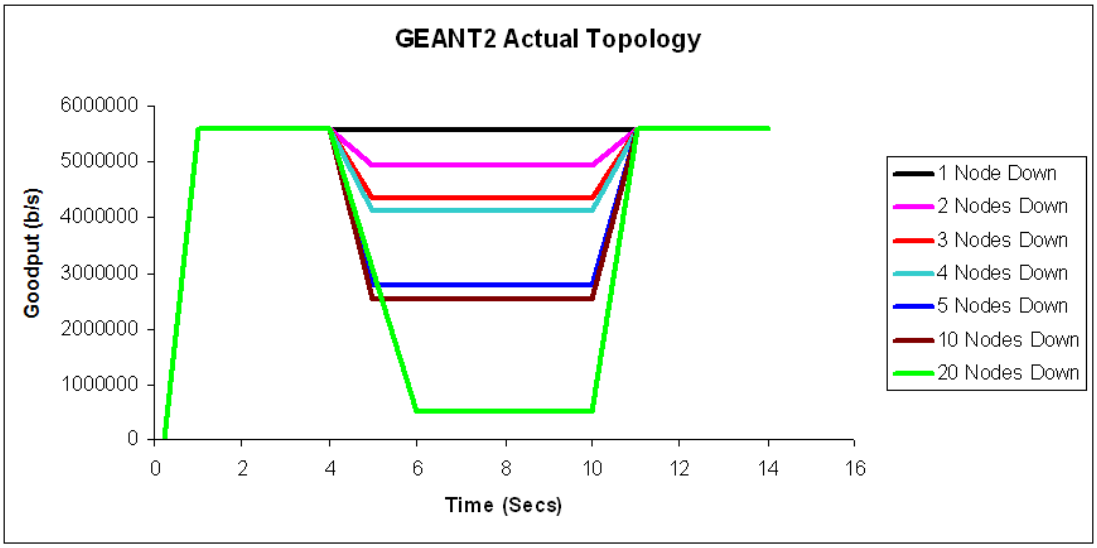


Figure 5.9. Random Node Failures of GÉANT2 Actual Topology

and 5.10 show the results for the GÉANT2 actual topology under random node outages with 1, 2, 3, 4, 5, 10, 20 nodes put down. The graphs clearly show more degradation of throughput with more nodes down. Similarly, the packet delivery ratio decreases with increasing number of nodes down. The effect of node shutdown is more severe as compared to the effect of link cuts because each node failure is equivalent to all of its connected link failing.

Figure 5.9 shows goodput for the GÉANT2 network with respect to node failure. The goodput decreases with the number of nodes down. Similarly the packet delivery ratio, shown in Figure 5.10, shows the expected decrease when nodes are randomly shut down.

Similar results are obtained for the Sprint network; Figure 5.11 shows goodput and Figure 5.10 shows the corresponding packet delivery ratios. With 20 random nodes shut down, the goodput for the Sprint network is worse than the GÉANT2 network because a greater proportion of the network has failed.

The same number of random nodes were shut down for the Sprint synthetic

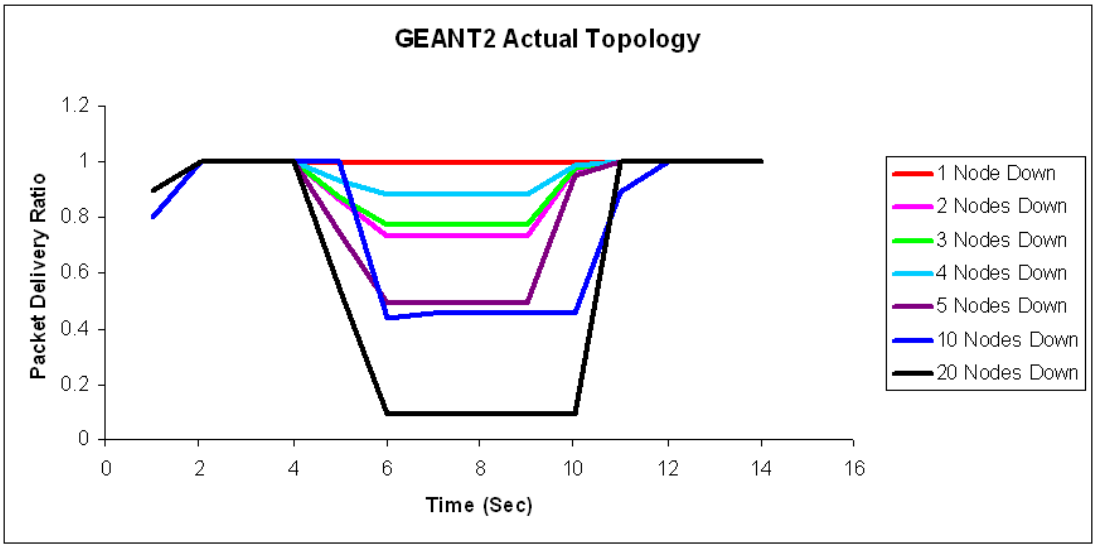


Figure 5.10. Random Node Failures of GÉANT2 Actual Topology

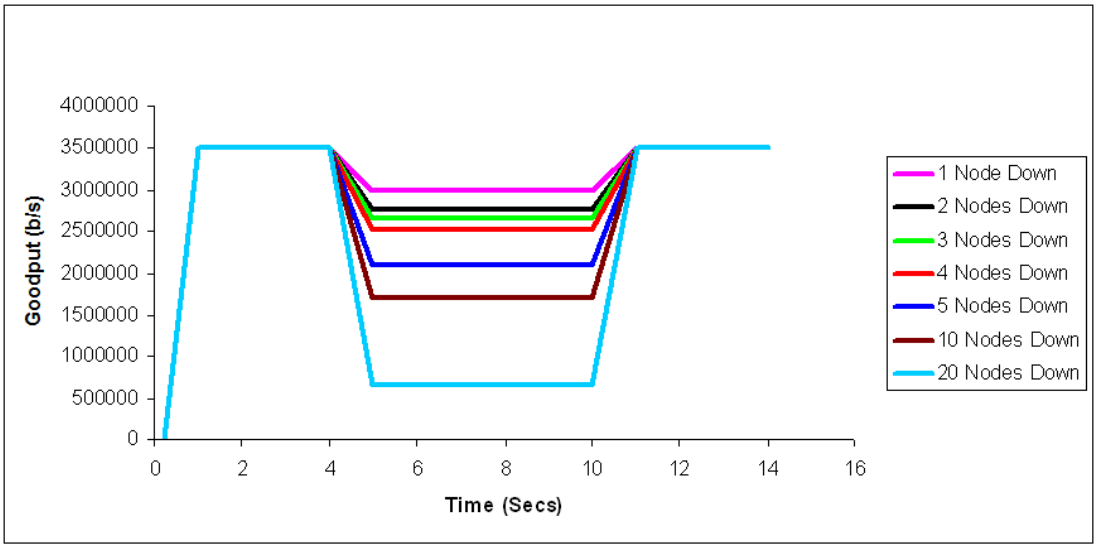


Figure 5.11. Random Node Failures of Sprint Actual Topology

resilient and fragile topologies. Their corresponding results were analysed against those for the Sprint actual topology. Figure 5.13 shows improved response over Figure 5.11 because of the more resilient topology. The same improvement is visible in the packet delivery ratio curves in Figure 5.14. On the other hand, con-

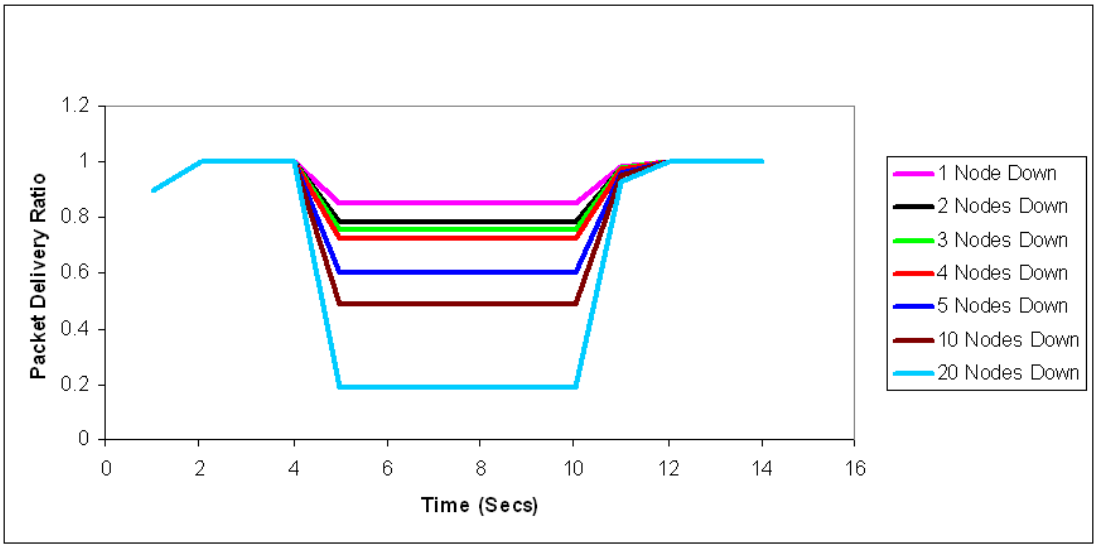


Figure 5.12. Random Node Failures of Sprint Actual Topology

siderable degradation is observed for the Sprint synthetic fragile topology goodput and packet delivery ratio as compared to the Sprint actual topology, as shown in Figures 5.15 and 5.16.

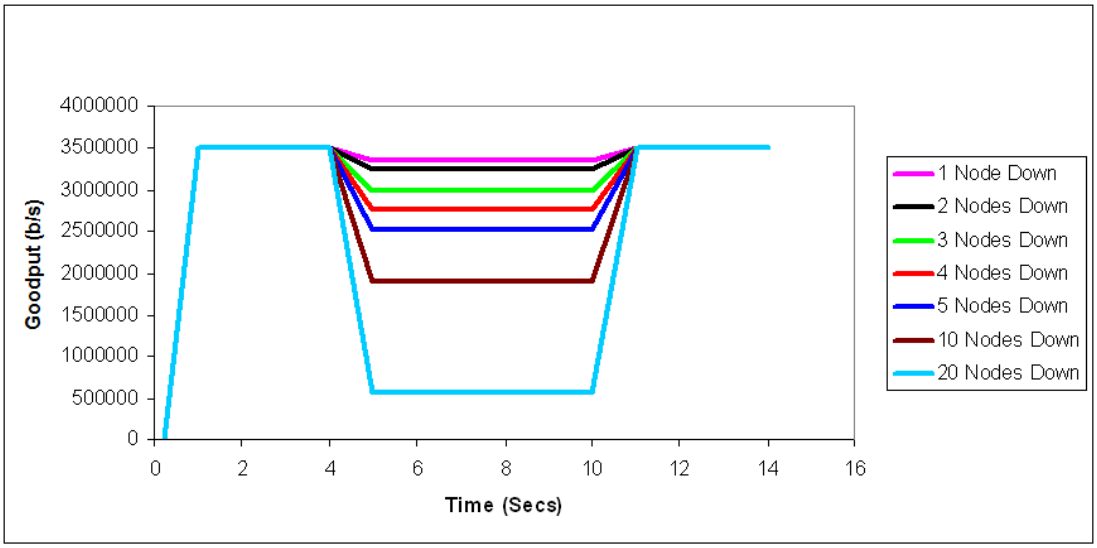


Figure 5.13. Random Node Failures of Sprint Resilient Topology

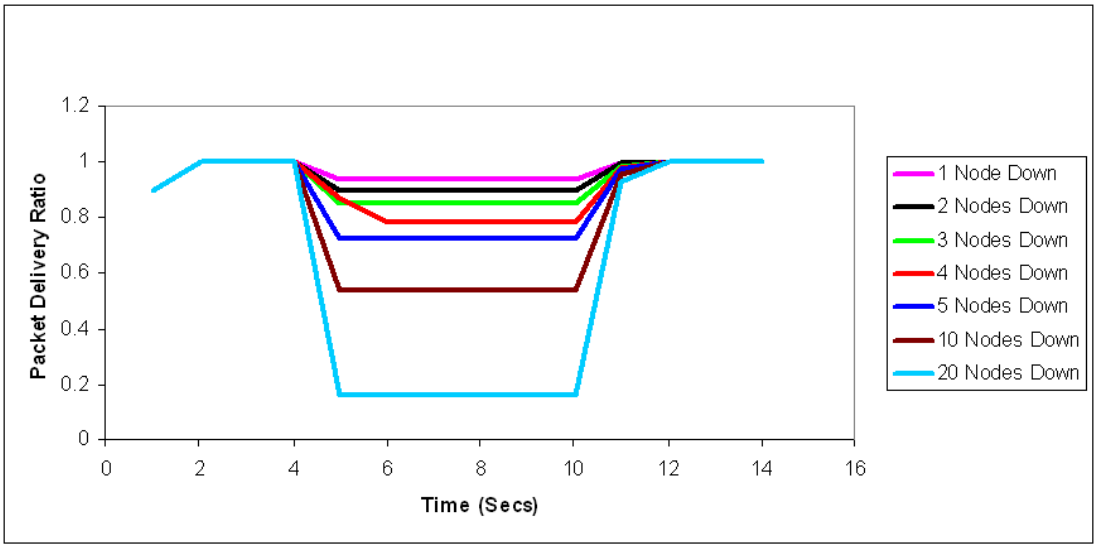


Figure 5.14. Random Node Failures of Sprint Resilient Topology

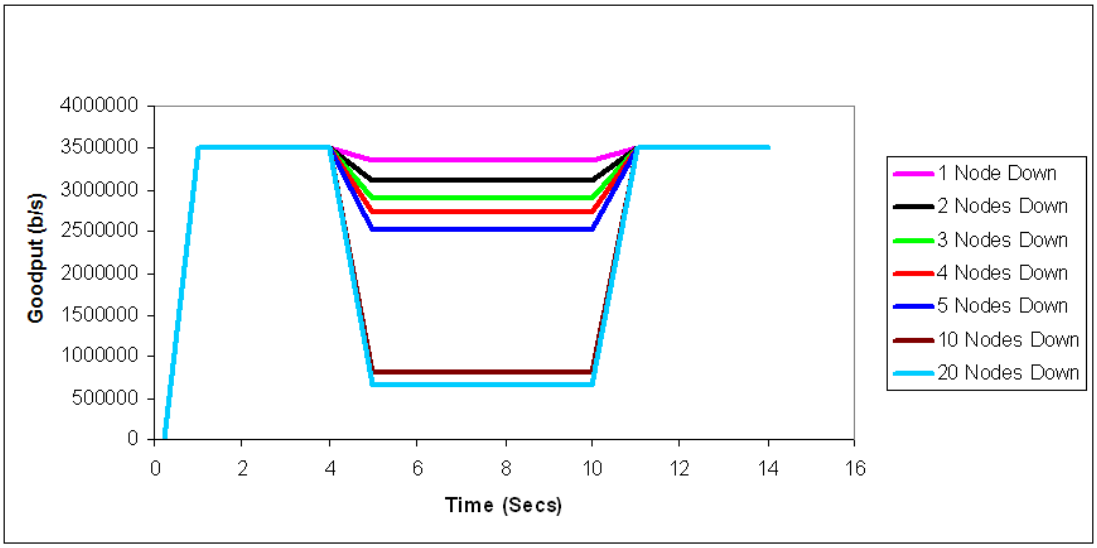


Figure 5.15. Random Node Failures of Sprint Fragile Topology

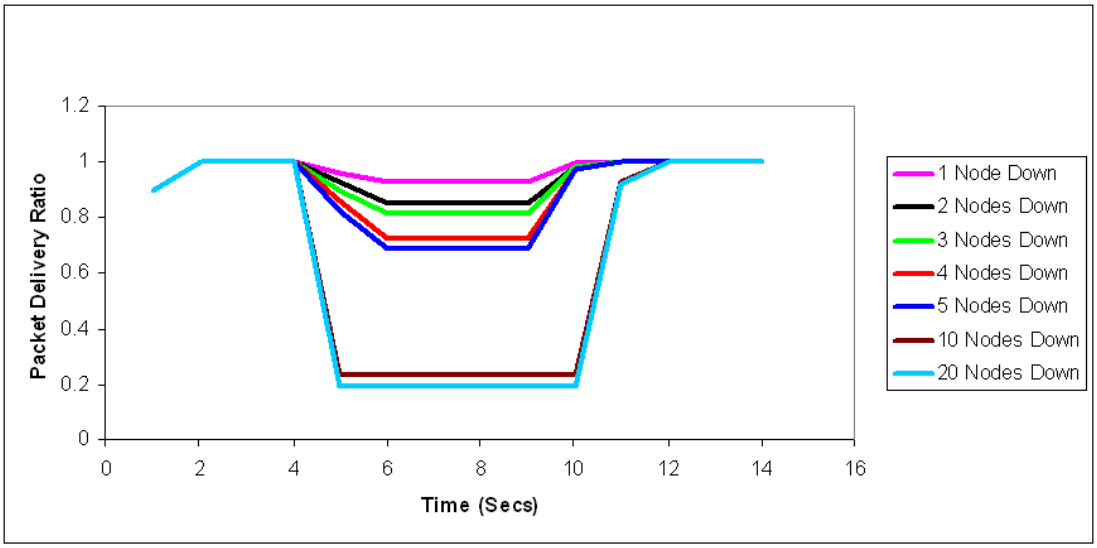


Figure 5.16. Random Node Failures of Sprint Fragile Topology

5.4 Resilience to Natural Disasters

Natural Disasters pose a significant challenge to network operation due to the large number of nodes and links that may be disrupted. Using the simulator designed for this thesis, a geographical study of a specific network is made possible. Now we can predict the behavior of a network under any large scale natural disaster. For the geographical scenario, the Sprint actual topology was studied and three areas were targetted as shown in Figure 5.17. It can be clearly seen in Figure 5.18 that the throughput under the challenge which covers the Florida region is better than the throughput under the West and the Northeast challenge. The West and the Northeast region both contain a major portion of the Sprint's network and hence they are more affected as compared to Florida. The Northeast challenge shows worse response because the number of node outages in the Northeast region are greater than the number of node outages in the West or Florida. Similar results are seen for the packet delivery ratio in Figure 5.19.

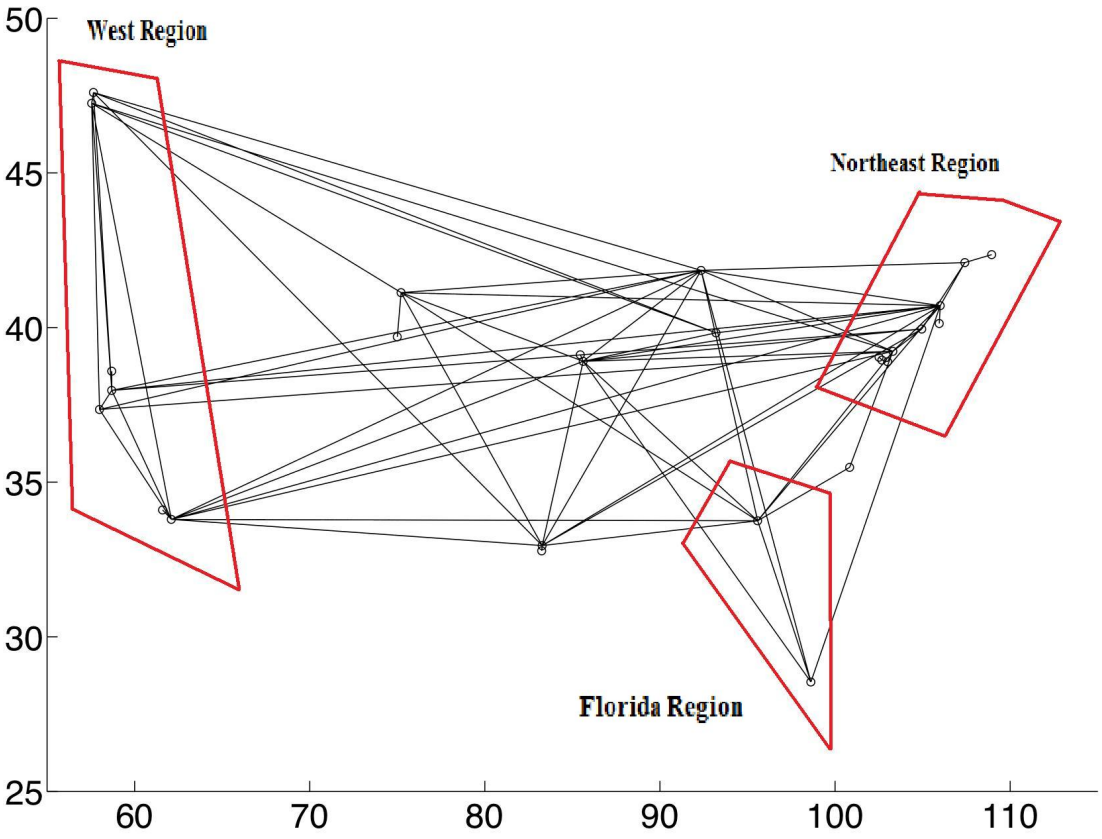


Figure 5.17. Sprint Geographical Attacks

5.5 Attack Scenarios

Attacking the fiber optic cables which constitute the links between the nodes or attacking the nodes themselves to switch them off are an attractive means to jeopardise the network for the adversaries, who can target only the key nodes or links to do the most damage to the network.

5.5.1 Link Attack

To compare the results for random link failures against the consequences of intelligent attacks against the network infrastructure, particular links were attacked and hence taken down. Figure 5.20 shows the links which were cut for the

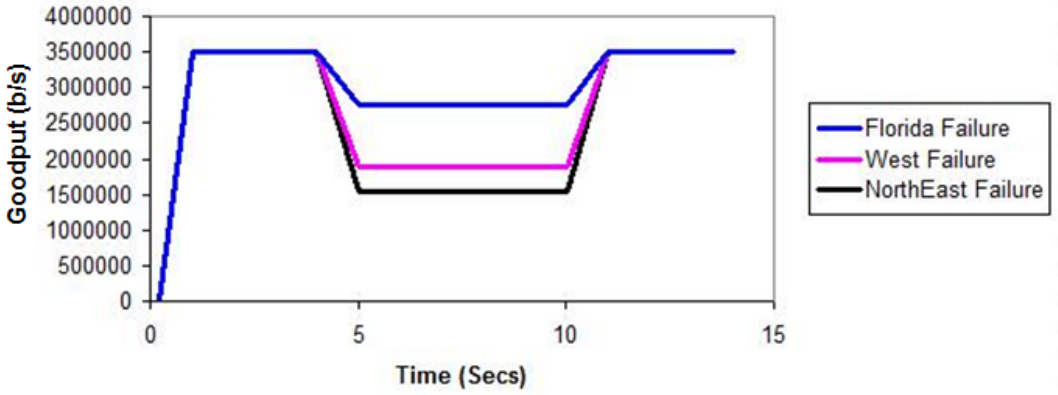


Figure 5.18. Geographical Shutdown Sprint Actual Topology

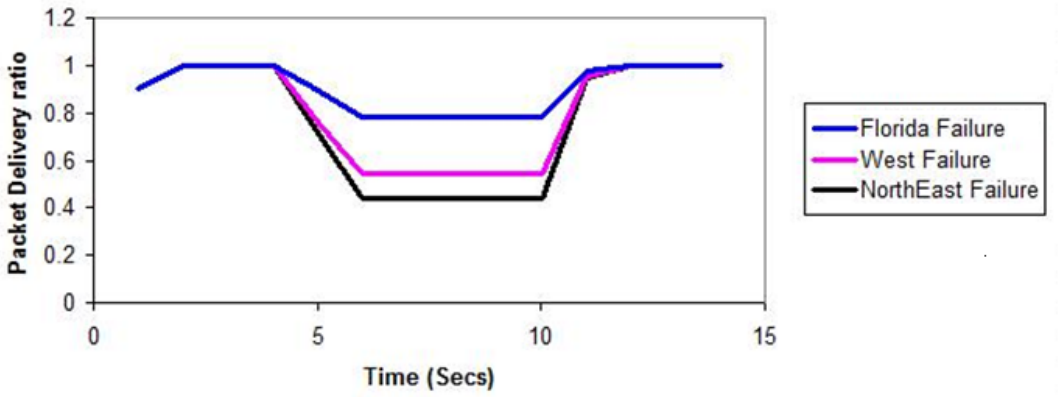


Figure 5.19. Geographical Shutdown Sprint Actual Topology

GÉANT2 network to observe network performability.

The GÉANT2 results show degraded response because the links that were removed partitioned the network. Figure 5.21 shows the goodput decrease and Figure 5.22 shows the corresponding packet delivery ratio decrease.

Figure 5.23 shows the links which were cut for the Sprint actual topology to observe the network behavior. In spite of attacking the five links shown in Fig-

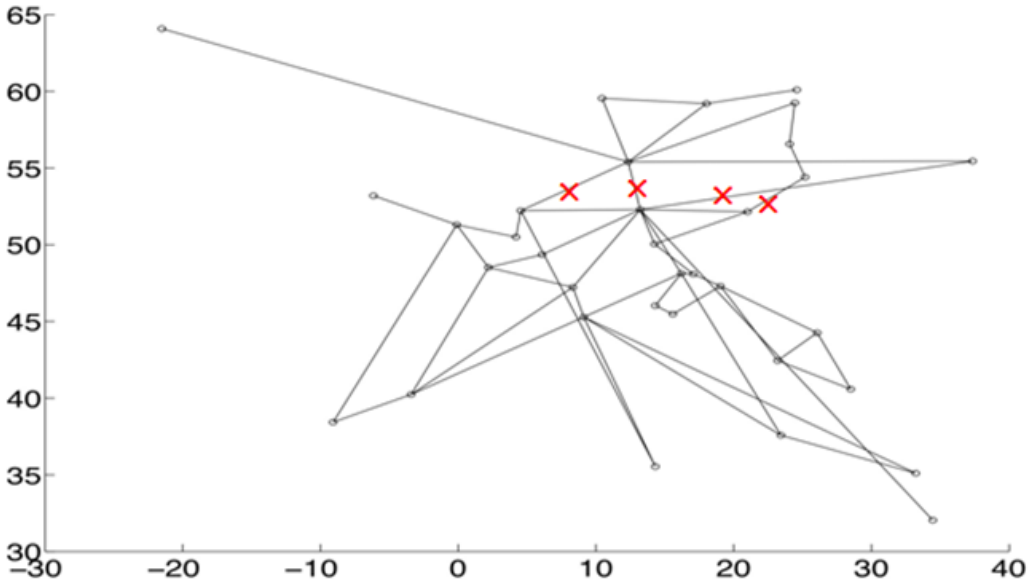


Figure 5.20. Link Attack Against GÉANT2 Actual Topology

ure 5.23, the throughput shown in Figure 5.24 and the packet delivery ratio shown in Figure 5.25 did not show considerable degradation because of the redundant paths present to permit rerouting the packets flowing in the network.

5.5.2 Node Attack

Figures 5.26 and 5.27 show the effects of attacks targetting nodes rather than links. The effect of the attack is evident in Figures 5.28 and 5.29. It is quite apparent that significant damage has been done while attacking only a few nodes.

Figures 5.30 and 5.31 depict attacks scenarios against the Sprint network topology. Key nodes are targetted that would inflict the most damage.

Figure 5.32 shows the goodput results when these three and five key nodes

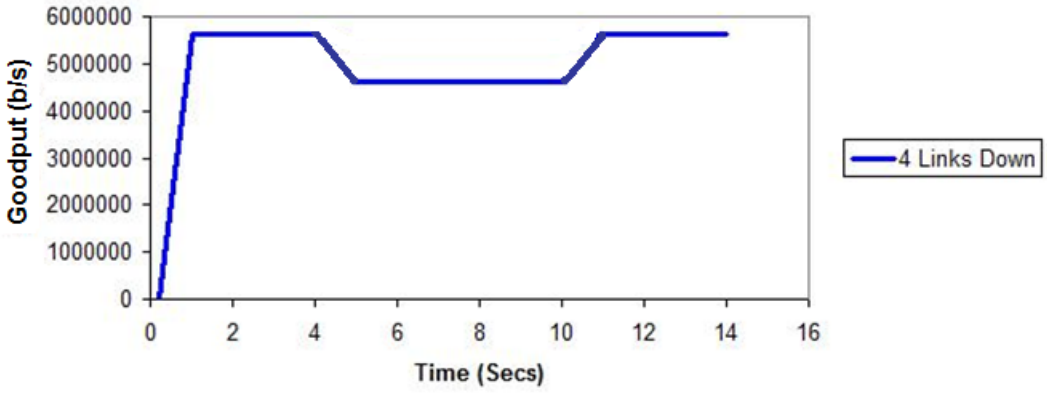


Figure 5.21. Link Attack Against GÉANT2 Actual Topology

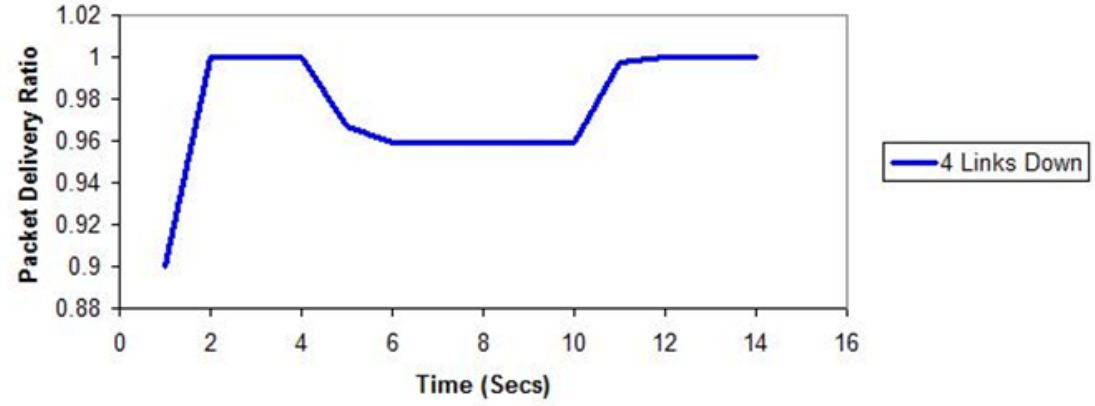


Figure 5.22. Link Attack Against GÉANT2 Actual Topology

are compromised. It is obvious that with three and five random nodes down, the goodput did not decrease to such a level as it did for three or five selectively attacked nodes. Similar results are seen for the packet delivery ratio under node attack, as shown in Figure 5.33.

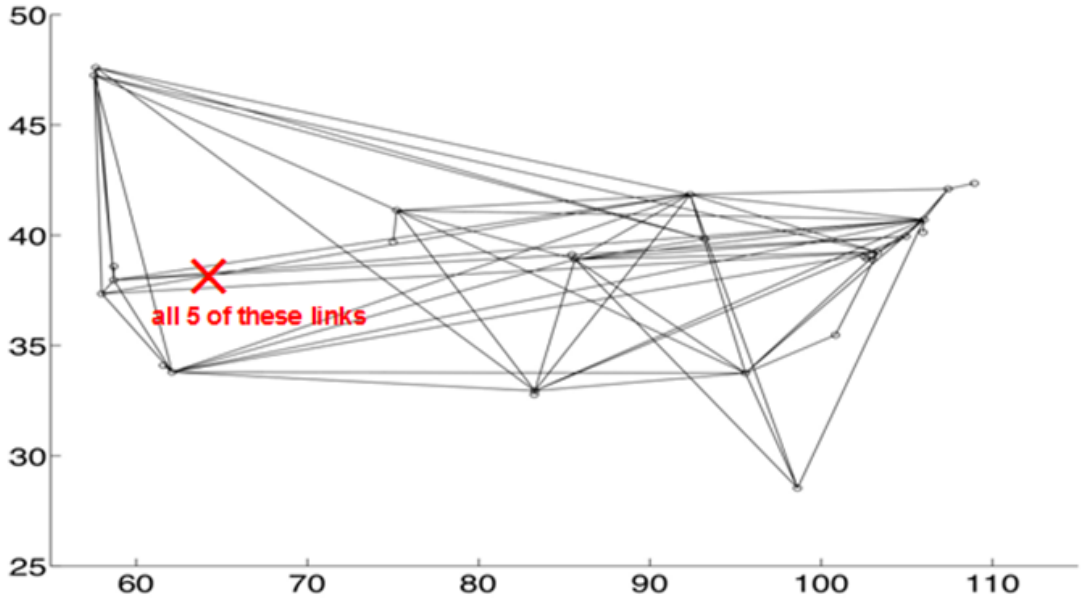


Figure 5.23. Link Attack Against Sprint Actual Topology

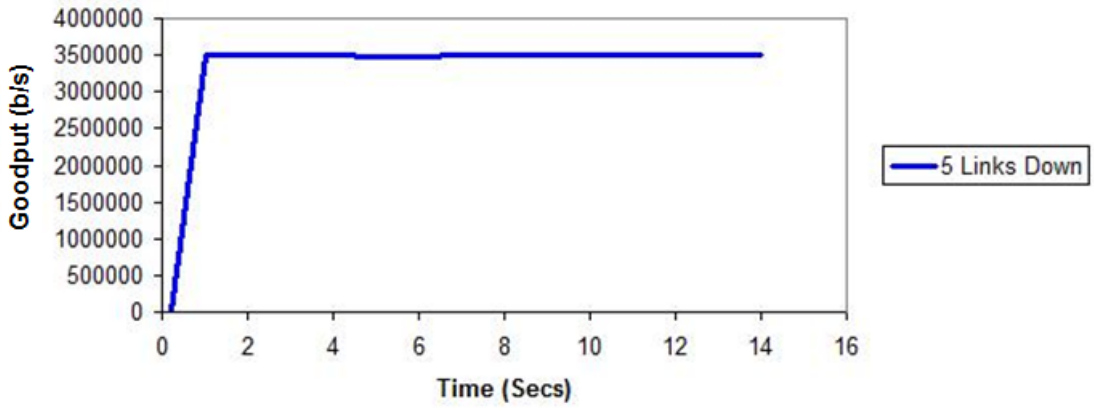


Figure 5.24. Link Attack Against Sprint Actual Topology

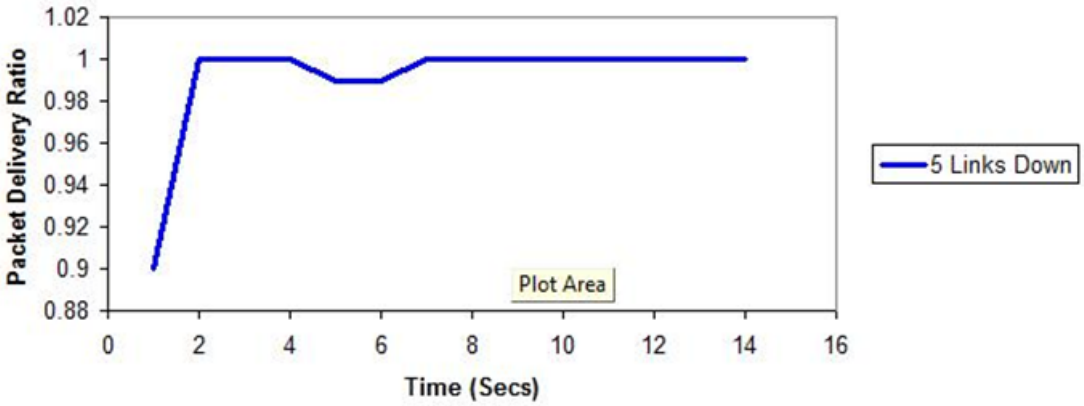


Figure 5.25. Link Attack Against Sprint Actual Topology

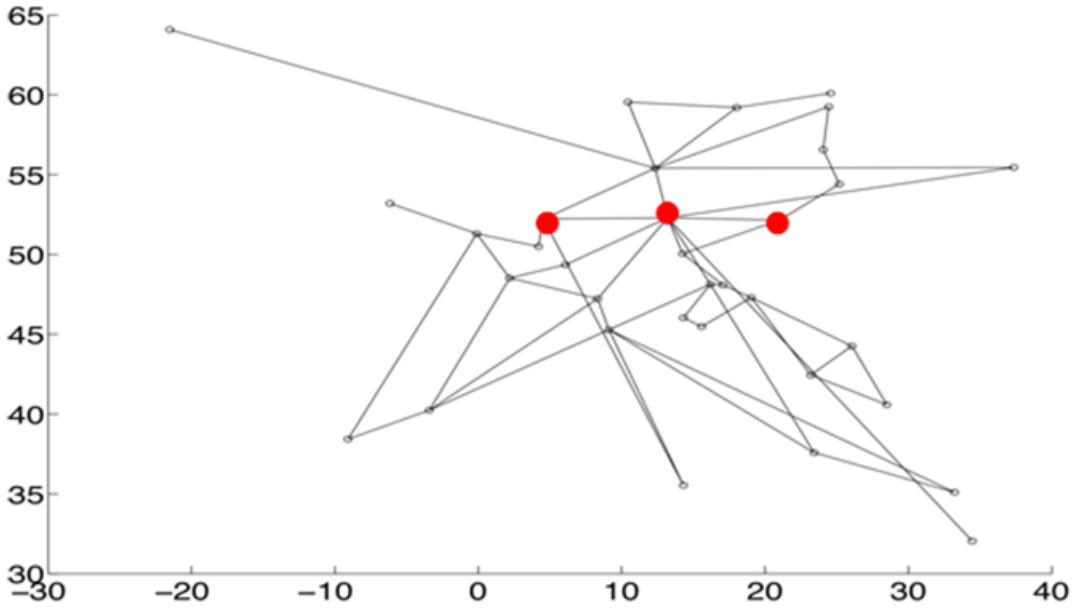


Figure 5.26. 3 Node Attack Against GÉANT2 Actual Topology

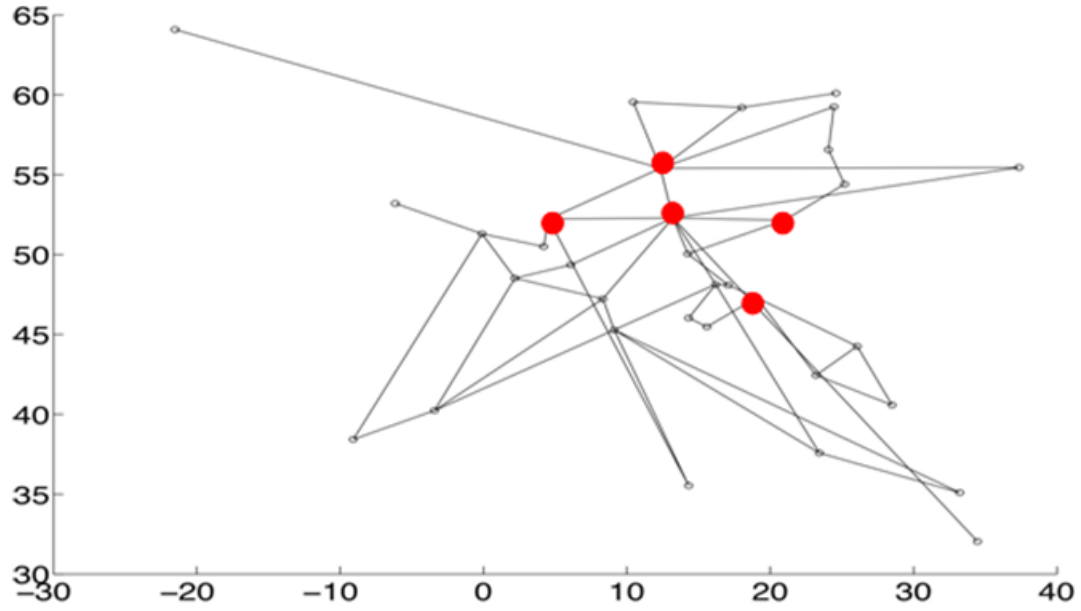


Figure 5.27. 5 Node Attack Against GÉANT2 Actual Topology

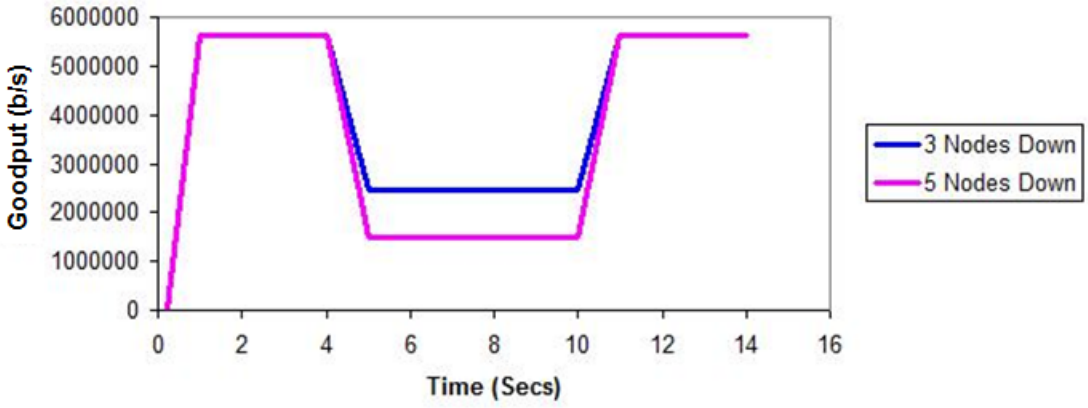


Figure 5.28. Node Attack Against GÉANT2 Actual Topology

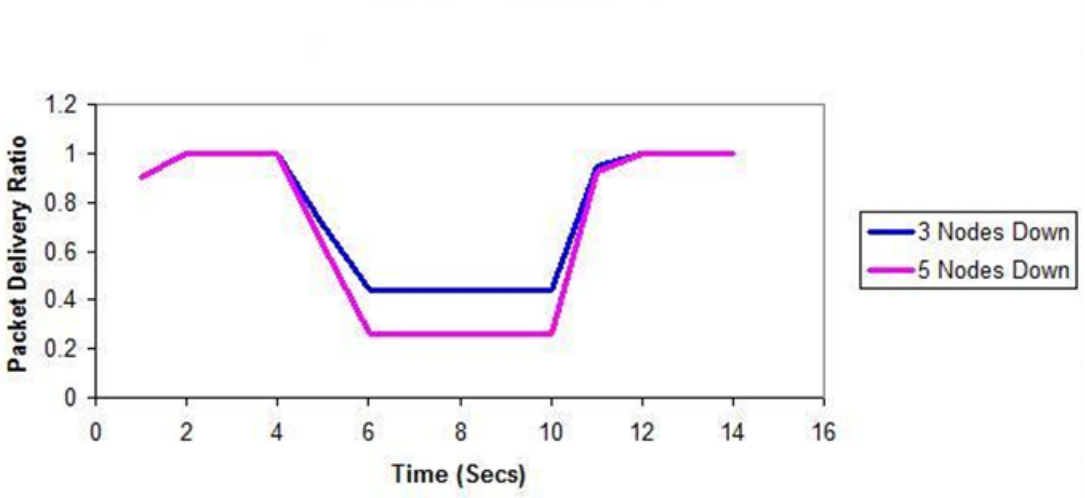


Figure 5.29. Node Attack Against GÉANT2 Actual Topology

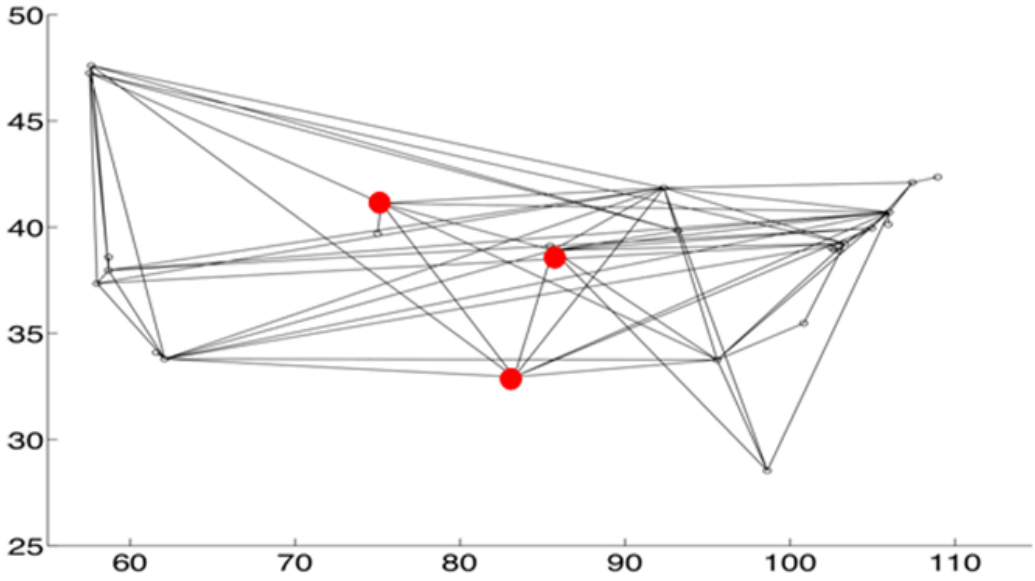


Figure 5.30. 3 Node Attack Against Sprint Actual Topology

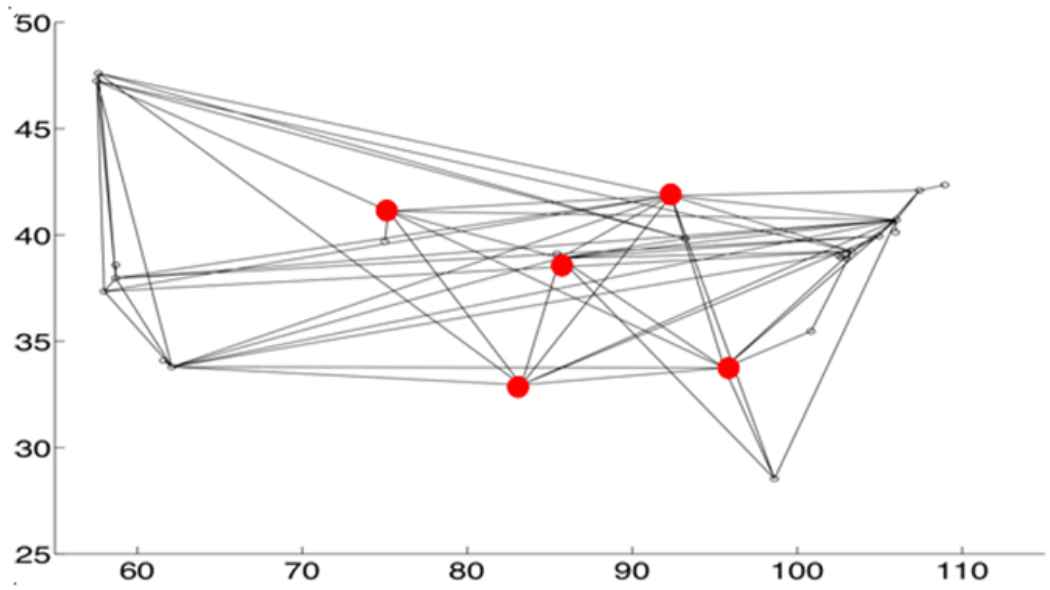


Figure 5.31. 5 Node Attack Against Sprint Actual Topology

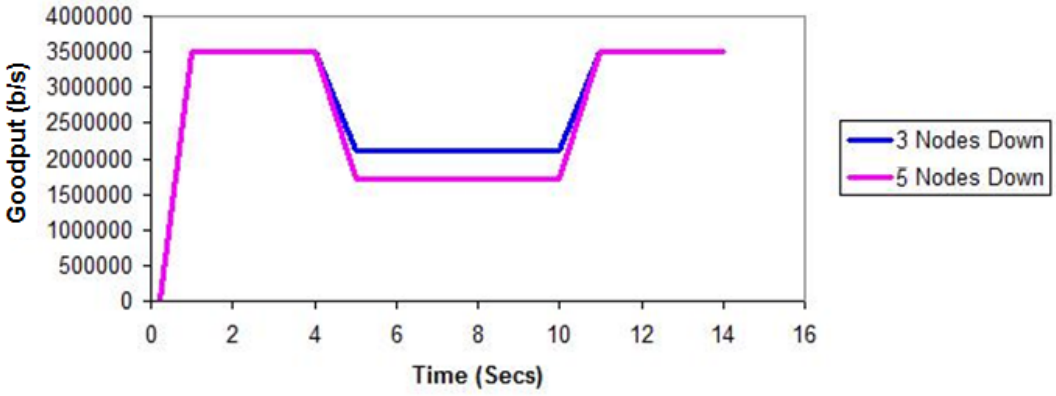


Figure 5.32. Node Attack Against Sprint Actual Topology

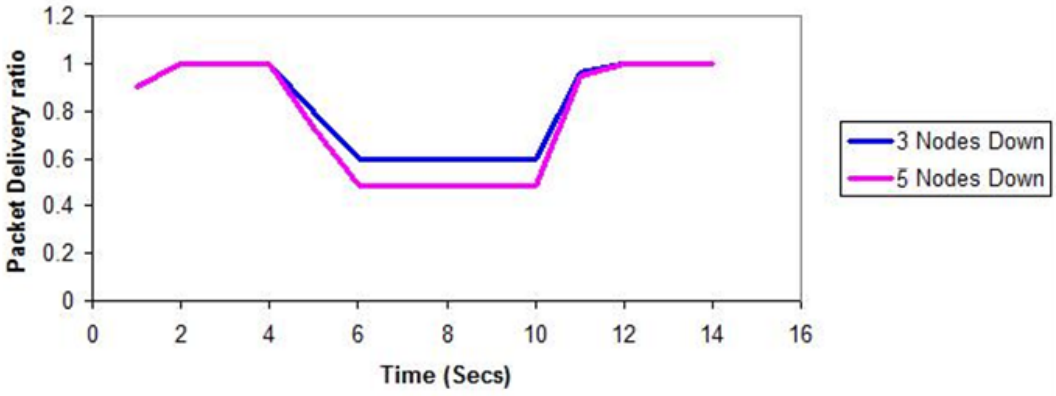


Figure 5.33. Node Attack Against Sprint Actual Topology

5.6 Attack and Failure Comparisons

After the individual scenarios were simulated and their results analysed, multiple scenarios are plotted against performance parameters to compare the performability and thus the resilience of the different network topologies. For comparison purposes, the GÉANT2 actual, Sprint actual, Sprint resilient, and Sprint fragile results were all plotted together. First we compare node failures, followed by link failures. The time step chosen for all of the comparison plots was 6 seconds, at which point the network was under the influence of a challenge. Figure 5.34 clearly shows a decrease in goodput with the increase of the number of random nodes shut down. Both the Sprint actual topology and Sprint synthetic resilient topology give approximately similar results, with the resilient topology showing a bit better value of goodput. The Sprint synthetic fragile topology, on the other hand, clearly shows degraded performance.

Figure 5.35 shows degradation with the Sprint synthetic resilient topology giving the best performance and the Sprint synthetic fragile topology giving the

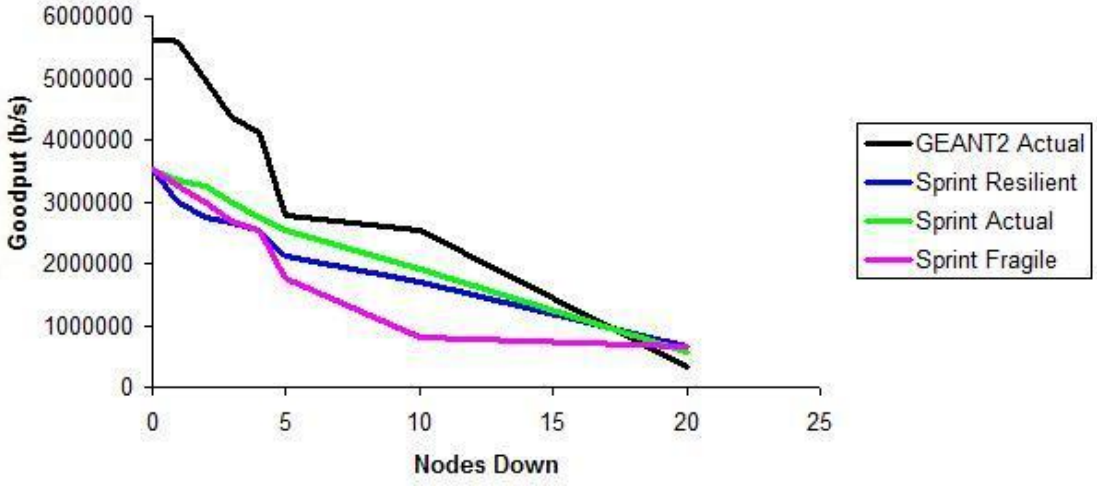


Figure 5.34. Goodput vs. Random Node Down

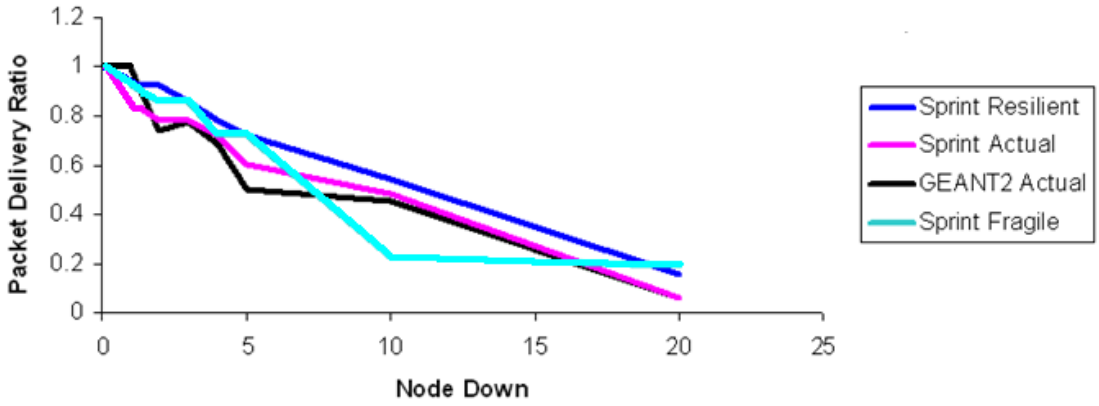


Figure 5.35. PDR vs. Random Node Down

worst packet delivery ratio. The curve for the GÉANT2 actual topology shows results very similar to the Sprint actual topology but cannot be directly compared since traffic is not normalised.

Figure 5.36 shows the value for the goodput when three and five nodes are attacked for both the Sprint actual and GÉANT2 actual topologies. GÉANT2 shows a steeper decline in performance due to its partitioning when more nodes

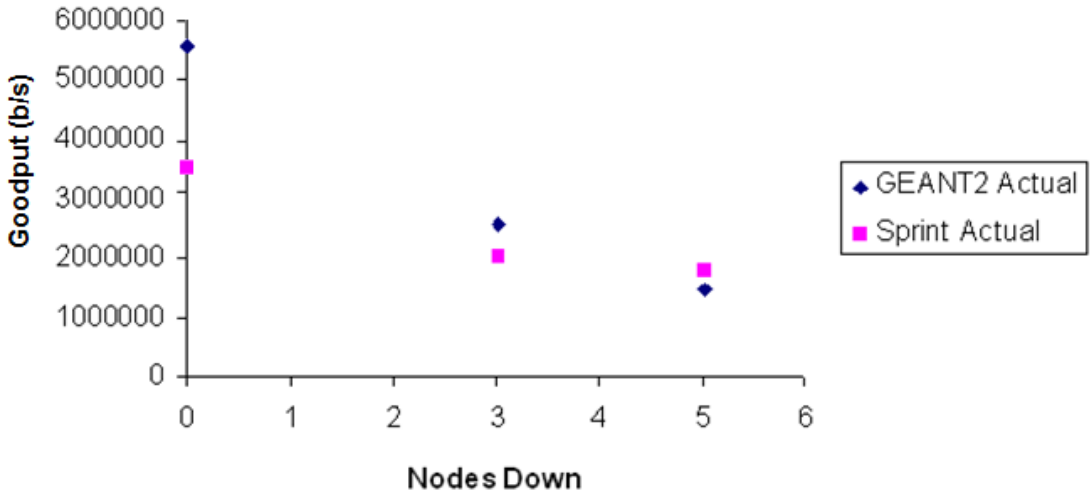


Figure 5.36. Goodput vs. Node Attack

fail.

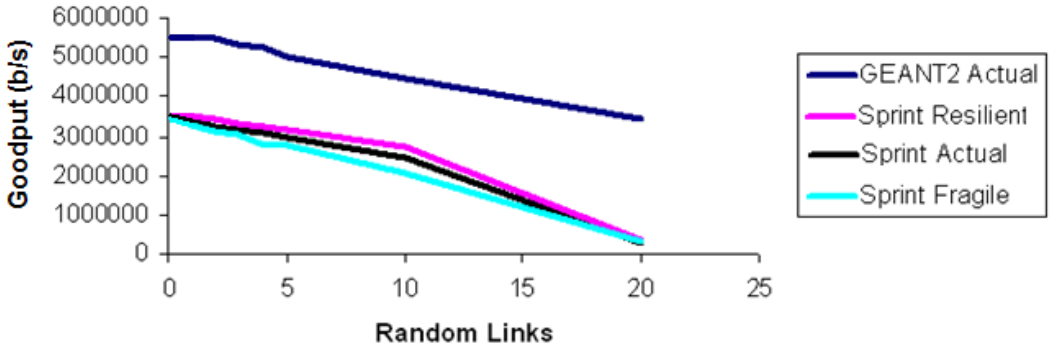


Figure 5.37. Goodput vs. Random Link Down

Figure 5.37 shows the impact of link failures for the GÉANT2 actual topology, Sprint synthetic resilient topology, Sprint actual topology, and Sprint synthetic fragile topology. As expected, the curves show slow degradation with an increasing

number of links down. The packet delivery ratio curves show similar results, as in Figure 5.38.

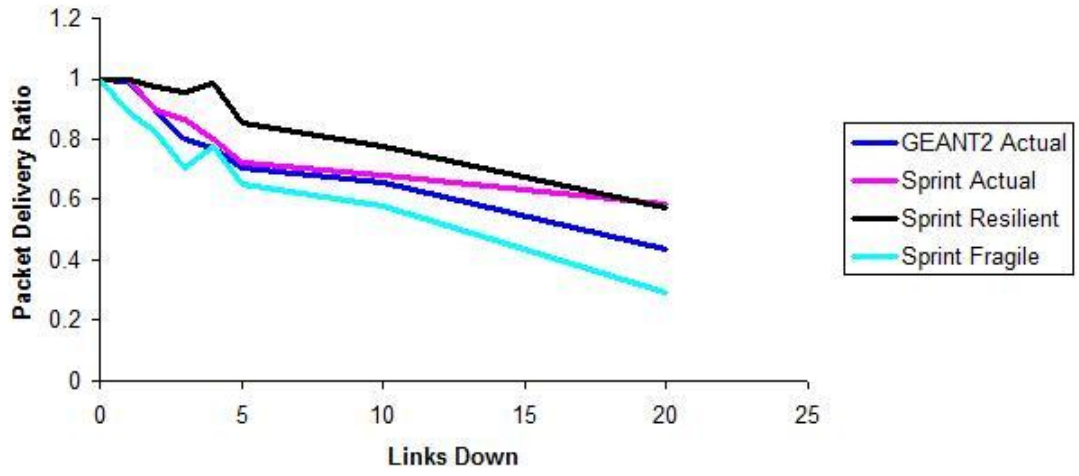


Figure 5.38. PDR vs. Random Link Down

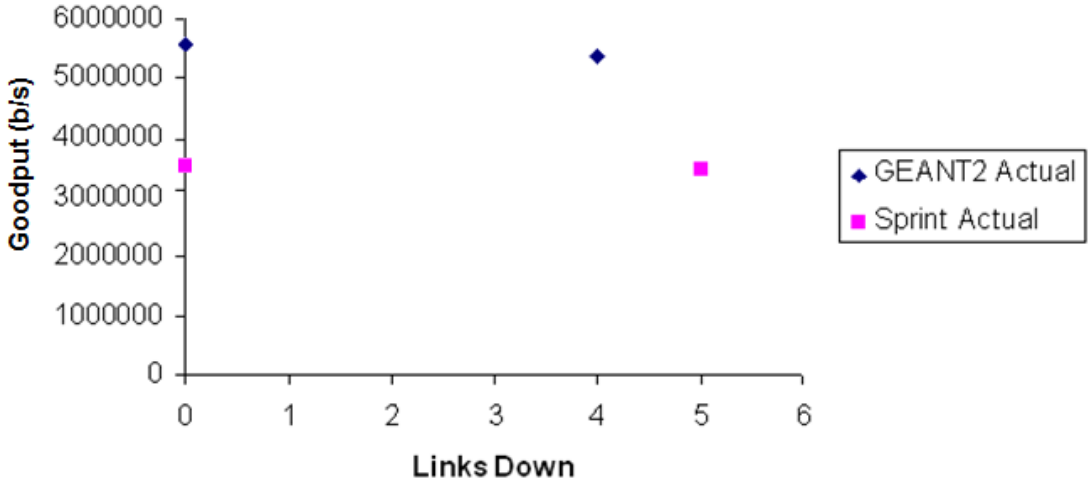


Figure 5.39. Goodput vs. Link Attack

Figure 5.39 shows the goodput values when four links were attacked for the GÉANT2 actual topology and five for the Sprint actual topology. These plots

show little degradation even though links went down due to sufficient alternate paths.

Chapter 6

Conclusions and Future Work

This chapter concludes this thesis summarising the simulator's strengths and weaknesses in the light of experimental results obtained.

6.1 Conclusions

Ns-3 is a newly developed network simulator that was used to analyse the resilience of network topologies to challenges in this thesis. The challenge module developed is unique as it applies generic challenges to any network model. Network specifications and challenge specifications are brought together by the simulator to model the resilience of the network under the challenge.

In designing this challenge simulator, a new methodology has been introduced for evaluating network resilience which is independent of network characteristics. In other words, a challenge can be applied to a network independent of its characteristics, and then its resilience can be evaluated.

By performing the tests on two different communication networks, Sprint and GÉANT2, the above mentioned concept is proved. We have seen that whenever

the network is either under some challenge including attack, the performance of the network decreases, and the decrease has been shown to reflect the severity of the challenge. Hence, this model can be used to evaluate resilience and compare the performability of different topologies.

6.2 Contributions

This thesis provides a new tool that permits the separartion of challenge specification from the network model, so that resilience can be simulated without modification of the network model for each challenge. Furthermore this thesis demonstrates the utility of this technique by evaluating the resilience of several network topologies under several challenges.

6.3 Future Work

This challenge simulator design has only considered wired networks. In the real world, wireless networks are increasingly important. Wireless networks consist of a set of untethered nodes that communicate with other nodes that lie in their transmission range and may be mobile.

Wireless links are a critical point of attack. A popular challenge includes node jammers that send signal with a very high power impacting the normal transmission. Link attenuation can also lead to impairment and hence a disconnected network. Incorporating node jammer and link attenuation challenges in the simulator design can further enhance its utility.

This simulator aims to evaluate the performability aspect of resilience of a network, but does not help to improve it. As a future extension of this simulator,

another module can be designed which takes the results of this simulator for a given network and provides input to the topology generator to generate more resilient topologies.

This thesis design has only focused on static challenges. Static challenges begin at some time t_0 over an area or random set of nodes and end at some other time t_1. On the other hand, a *dynamic* challenge allows the evolution of the challenge over time and network topology. This simulator can be extended to permit the specification of a trajectory of the attack scenario polygon.

Bibliography

[1] Additions to the Ns-2 network simulator to handle the wireless functionality. http://www.ece.cmu.edu/wireless/.

[2] Dependability. https://wiki.ittc.ku.edu/resilinets_wiki/index.php/Dependability.

[3] Disruption Tolerance. https://wiki.ittc.ku.edu/resilinets_wiki/index.php/Disruption_Tolerance.

[4] Fault Tolerance. https://wiki.ittc.ku.edu/resilinets_wiki/index.php/Fault_Tolerance.

[5] Parallel Simulation Environment for Complex Systems (PARSEC). http://pcl.cs.ucla.edu/projects/parsec/.

[6] Performability. https://wiki.ittc.ku.edu/resilinets_wiki/index.php/Performability.

[7] Performability. http://www.doc.ic.ac.uk/~nd/surprise_95/journal/vol4/eaj2/report.html.

[8] Resilience. https://wiki.ittc.ku.edu/resilinets_wiki/index.php/Resilience.

[9] Robustness. https://wiki.ittc.ku.edu/resilinets_wiki/index.php/Robustness.

[10] Security. https://wiki.ittc.ku.edu/resilinets_wiki/index.php/Security.

[11] Survivability. https://wiki.ittc.ku.edu/resilinets_wiki/index.php/Survivability.

[12] GÉANT2. http://www.geant2.net/.

[13] NAM: Network Animator. http://www.isi.edu/nsnam/nam/.

[14] OPNET Modeler, Accelerating Network R&D. http://www.opnet.com/solutions/network_rd/modeler.html.

[15] Traffic Tolerance. https://wiki.ittc.ku.edu/resilinets_wiki/index.php/Traffic_Tolerance.

[16] Object Oriented Tcl. http://bmrc.berkeley.edu/research/cmt/cmtdoc/otcl/tutorial.html, 1995.

[17] A Comprehensive GloMoSim Tutorial. http://www.ccs.neu.edu/course/csg250/Glomosim/glomoman.pdf, March 2004.

[18] GT-ITM. http://www.cc.gatech.edu/projects/gtitm/, 2007.

[19] The Ns-2 Network Simulator. http://www.isi.edu/nsnam/ns/doc/ns_doc.pdf, January 2009.

[20] The Ns-3 network simulator. http://www.nsnam.org/docs/release/tutorial.pdf, July 2009.

[21] OPNET Technologies. http://www.opnet.com/, 2009.

[22] Algirdas Avizienis. Fault-tolerance and fault-intolerance: Complementary approaches to reliable computing. In *ACM SIGPLAN*, volume 10, pages 458–464, 1975.

[23] Algirdas Avizienis, Jean-Claude Laprie, Brian Randell, and Carl Landwehr. Basic concepts and taxonomy of dependable and secure computing. In *IEEE Transactions on Dependable and Secure Computing*, volume 1, pages 11–33, January - March 2004.

[24] Xiangqian Chen, K. Makki, Kang Yen, and N. Pissinou. A new network topology evolution generator based on traffic increase and distribution model. In *Networking, 2007. ICN '07. Sixth International Conference*, page 56. IEEE Computer Society, April 2007.

[25] R.J. Ellison, D.A. Fisher, R.C. Linger, H.F. Lipson, T. Longstaff, and N.R. Mead. Survivable Network Systems: An Emerging Discipline. Technical Report CMU/SEI-97-TR-013, Carnegie Mellon University, Pittsburg, P.A., November 1997.

[26] F.P. Fontan, A. Nunez, A. Volcarce, and U.C. Fiebig. Converting simulated rain-rate series into attenuation series using the synthetic storm technique. In *Proceedings of the Cost 280 3rd International Workshop*, June 2005.

[27] Sanjay Goel, Salvatore Belardo, and Laura Iwan. A resilient network that can operate under duress: To support communication between government agencies during crisis situation. In *37th Hawaii International Conference*

on System Sciences (HICSS'04), volume 5, page 50123.1. IEEE Computer Society, Washington, DC, USA, 2004.

[28] Abdul Jabbar, Qian Shi, Egemen Cetinkaya, and James P.G. Sterbenz. KU-LocGen: Location and Cost Constrained Network Topology Generator. Technical Report ITTC-FY2009-TR-45030-01, University of Kansas, December 2008.

[29] Alberto Medina, Anukool Lakhina, Ibrahim Matta, and John Byers. BRITE: Universal Topology Generation from a User's Perspective. Technical Report BUCS-TR-2001-003, BOSTON UNIVERSITY, April 2001.

[30] Peter G. Neumann. System and Network Trustworthiness in Perspective. In *Proceedings of the 13th ACM conference on Computer and communications security*, pages 1–5, October - November 2006.

[31] Jorge Nuevo. A Comprehensive GloMoSim Tutorial. `http://www.ccs.neu.edu/course/csg250/Glomosim/glomoman.pdf`, March 2004.

[32] F. Permadi. Ray Casting Tutorial. `http://www.permadi.com/tutorial/raycast/index.html`.

[33] Neil Spring, Ratul Mahajan, David Wetherall, and Thomas Anderson. Measuring ISP topologies with Rocketfuel. In *ACM SIGCOMM Computer Communication Review*, volume 32, pages 133–145, October 2002.

[34] James P.G. Sterbenz and David Hutchison. ResiliNets. `https://wiki.ittc.ku.edu/resilinets_wiki/index.php/Main_Page`, 2006 - 2008.

[35] B.M. Waxman. Waxman Random Network Topology Generator. `http://www.math.uu.se/research/telecom/software/stgraphs.html`.

[36] B.M. Waxman. Routing of Multipoint Connections. In *Selected Areas in Communications, IEEE Journal*, volume 6, pages 1617–1622, December 1988.